



Universidad de Valladolid

Facultad de Ciencias

TRABAJO FIN DE GRADO

Grado en Matemáticas

**Integración numérica de sistemas
diferenciales de segundo orden.**

Alumna: Beatriz Alonso Román

Tutora: M.P. Calvo Cabrero

Índice general

Introducción	3
1 Métodos Runge-Kutta-Nyström (RKN)	5
1.1 Definición	5
1.2 Condiciones de orden	8
1.3 Estabilidad lineal	26
1.4 Estimación del error local y cambio de paso	31
2 Integración numérica del problema de Kepler	35
2.1 Descripción del problema	35
2.2 Métodos utilizados	36
2.3 Resultados numéricos	38
Bibliografía	47
A. Programas de Matlab	49
A.1 Implementación de paso fijo	49
A.2 Implementación con paso variable	51
A.3 Estabilidad del método RKN4(3)4FM	53
A.4 Otras funciones auxiliares	55
B. Caracterización de la región de estabilidad absoluta	57

Introducción

El objetivo de este trabajo es el estudio de los métodos Runge-Kutta-Nyström para la integración numérica de sistemas de ecuaciones diferenciales ordinarias de segundo orden. Dentro del Grado en Matemáticas, en la asignatura obligatoria del tercer curso, *Ampliación de Análisis Numérico*, estudiamos una muy breve introducción a la resolución numérica de sistemas diferenciales de primer orden mediante métodos Runge-Kutta y métodos lineales multipaso. Dichos conocimientos son ampliados en la asignatura optativa de cuarto curso, *Solución Numérica de Ecuaciones Diferenciales*. El trabajo de Fin de Grado que presentamos se apoya en los conocimientos adquiridos en estas asignaturas, pero desarrolla una teoría específica para el tratamiento numérico de ecuaciones diferenciales de segundo orden, que es completamente nueva para el alumno.

En el primer capítulo, se ha hecho un estudio detallado de los métodos Runge-Kutta-Nyström que incluye la construcción de las correspondientes condiciones de orden, el análisis de la estabilidad lineal, así como la estimación del error local y la implementación eficiente del cambio de paso durante la integración.

En el segundo capítulo, se han utilizado métodos de la familia estudiada en el Capítulo 1 para la integración numérica del problema de Kepler. Este problema es uno de los problemas test que se han utilizado desde los años 70 del siglo XX para analizar y comparar la eficiencia de diferentes integradores de ecuaciones diferenciales cuando se implementan con paso variable. Hemos analizado la eficiencia de dos métodos de órdenes 4 y 6, respectivamente, que se han implementado tanto con paso fijo como con paso variable.

En el primer apéndice incluimos los programas que hemos implementado en Matlab, con los que se han obtenido los resultados y gráficas que hemos presentado y analizado en el Capítulo 2. Incluimos también los programas

usados para estudiar la estabilidad del método de orden 4.

En el segundo apéndice hemos incluido una caracterización alternativa de la región de estabilidad absoluta de un método Runge-Kutta-Nyström.

Por último, me gustaría agradecer a mi tutora, la Dra. Mari Paz Calvo Cabrero, por su dedicación, su tiempo y su paciencia en la elaboración de este Trabajo de Fin de Grado. Además, quisiera dar las gracias también a mi familia por su apoyo incondicional durante todo este tiempo.

Valladolid a 12 de Marzo de 2018

Capítulo 1

Métodos Runge-Kutta-Nyström (RKN)

1.1. Definición

Consideramos el sistema de ecuaciones diferenciales de segundo orden:

$$\begin{cases} \mathbf{y}''(t) = \mathbf{f}(\mathbf{y}(t)), \\ \mathbf{y}(t_0) = \mathbf{y}_0, \\ \mathbf{y}'(t_0) = \mathbf{v}_0, \end{cases} \quad (1.1)$$

con $\mathbf{y}_0, \mathbf{v}_0 \in \mathbb{R}^D$ y $\mathbf{f} : \mathbb{R}^D \rightarrow \mathbb{R}^D$ suficientemente regular y siendo $\mathbf{y}(t) = (y^1(t), \dots, y^D(t))^T$. Dadas aproximaciones \mathbf{y}_n y \mathbf{v}_n a la solución de (1.1) y a su derivada en tiempo $t = t_n$, se puede calcular una aproximación a la solución de (1.1) en tiempo $t_{n+1} = t_n + h$ aplicando un método Runge-Kutta-Nyström de s etapas del siguiente modo

$$\mathbf{k}_i = \mathbf{f}(\mathbf{y}_n + h\gamma_i\mathbf{v}_n + h^2 \sum_{j=1}^s \alpha_{ij}\mathbf{k}_j), \quad 1 \leq i \leq s, \quad (1.2)$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{v}_n + h^2 \sum_{i=1}^s \beta_i\mathbf{k}_i, \quad (1.3)$$

$$\mathbf{v}_{n+1} = \mathbf{v}_n + h \sum_{i=1}^s b_i\mathbf{k}_i, \quad (1.4)$$

donde $\boldsymbol{\gamma} = [\gamma_1, \dots, \gamma_s]^T$, $\mathbf{b} = [b_1, \dots, b_s]^T$, $\boldsymbol{\beta} = [\beta_1, \dots, \beta_s]^T$ y $\alpha = (\alpha_{ij})_{i,j=1}^s$ son los coeficientes que definen el método, y que se pueden representar en el tablero de Butcher asociado al método como

$$\begin{array}{c|c} \gamma & \alpha \\ \hline & \beta^T \\ \hline & \mathbf{b}^T \end{array}.$$

En el caso de que α sea una matriz estrictamente triangular inferior se dice que estamos ante un método explícito (en caso contrario se habla de método implícito) y se puede reescribir como

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_n + h\gamma_1 \mathbf{v}_n), \\ \mathbf{k}_i &= \mathbf{f}\left(\mathbf{y}_n + h\gamma_i \mathbf{v}_n + h^2 \sum_{j=1}^{i-1} \alpha_{ij} \mathbf{k}_j\right), \quad 2 \leq i \leq s, \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + h\mathbf{v}_n + h^2 \sum_{i=1}^s \beta_i \mathbf{k}_i, \\ \mathbf{v}_{n+1} &= \mathbf{v}_n + h \sum_{i=1}^s b_i \mathbf{k}_i. \end{aligned}$$

Notemos que la ecuación diferencial más general $\mathbf{y}''(t) = \mathbf{f}(t, \mathbf{y}(t))$, con $\mathbf{f} : \mathbb{R} \times \mathbb{R}^D \rightarrow \mathbb{R}^D$ suficientemente regular, se puede expresar como (1.1) definiendo $\mathbf{Y} = (\mathbf{y}^T, t)^T$ y $\mathbf{F} = (\mathbf{f}^T, 0)^T$ y teniendo en cuenta que

$$\mathbf{Y}''(t) = \begin{bmatrix} \mathbf{y}(t) \\ t \end{bmatrix}'' = \begin{bmatrix} \mathbf{f}(t, \mathbf{y}(t)) \\ 0 \end{bmatrix} = \mathbf{F}(\mathbf{Y}(t)).$$

Por otro lado, podemos representar el sistema diferencial de segundo orden (1.1) como un sistema diferencial de primer orden en \mathbb{R}^{2D}

$$\mathbf{Y}'(t) = \begin{bmatrix} \mathbf{y}(t) \\ \mathbf{y}'(t) \end{bmatrix}' = \begin{bmatrix} \mathbf{y}'(t) \\ \mathbf{f}(\mathbf{y}(t)) \end{bmatrix} = \mathbf{F}(\mathbf{Y}(t)) \quad (1.5)$$

introduciendo $\mathbf{Y} = (\mathbf{y}^T, \mathbf{y}'^T)^T$ y $\mathbf{F} = (\mathbf{y}'^T, \mathbf{f}(\mathbf{y})^T)^T$, y (1.5) se puede resolver numéricamente aplicando un método Runge-Kutta con coeficientes $\mathbf{b} = [b_1, \dots, b_s]^T$ y $A = (a_{ij})_{i,j=1}^s$. Las ecuaciones para avanzar un paso de longitud

h partiendo de $\mathbf{y}(t_0) = \mathbf{y}_0$, $\mathbf{y}'(t_0) = \mathbf{v}_0$ son

$$\mathbf{k}_i = \mathbf{v}_0 + h \sum_{j=1}^s a_{ij} \mathbf{k}'_j, \quad 1 \leq i \leq s, \quad (1.6)$$

$$\mathbf{k}'_i = \mathbf{f}\left(\mathbf{y}_0 + h \sum_{j=1}^s a_{ij} \mathbf{k}_j\right), \quad 1 \leq i \leq s, \quad (1.7)$$

$$\mathbf{y}_1 = \mathbf{y}_0 + h \sum_{i=1}^s b_i \mathbf{k}_i, \quad (1.8)$$

$$\mathbf{v}_1 = \mathbf{v}_0 + h \sum_{i=1}^s b_i \mathbf{k}'_i. \quad (1.9)$$

Si sustituimos (1.6) en (1.7) y (1.8) obtenemos

$$\begin{aligned} \mathbf{k}'_i &= \mathbf{f}\left(\mathbf{y}_0 + h \sum_{j=1}^s a_{ij} \left(\mathbf{v}_0 + h \sum_{l=1}^s a_{jl} \mathbf{k}'_l\right)\right) \\ &= \mathbf{f}\left(\mathbf{y}_0 + h \left(\sum_{j=1}^s a_{ij}\right) \mathbf{v}_0 + h^2 \sum_{l=1}^s \left(\sum_{j=1}^s a_{ij} a_{jl}\right) \mathbf{k}'_l\right), \end{aligned} \quad (1.10)$$

$$\begin{aligned} \mathbf{y}_1 &= \mathbf{y}_0 + h \sum_{i=1}^s b_i \left(\mathbf{v}_0 + h \sum_{j=1}^s a_{ij} \mathbf{k}'_j\right) \\ &= \mathbf{y}_0 + h \left(\sum_{i=1}^s b_i\right) \mathbf{v}_0 + h^2 \sum_{i,j=1}^s b_i a_{ij} \mathbf{k}'_j \\ &= \mathbf{y}_0 + h \mathbf{v}_0 + h^2 \sum_{j=1}^s \left(\sum_{i=1}^s b_i a_{ij}\right) \mathbf{k}'_j, \end{aligned} \quad (1.11)$$

de lo que deducimos que la aplicación de un método Runge-Kutta a (1.5) no es otra cosa que la aplicación del método Runge-Kutta-Nyström con coeficientes $\gamma_i = \sum_{j=1}^s a_{ij}$, $\alpha_{il} = \sum_{j=1}^s a_{ij} a_{jl}$ y $\beta_j = \sum_{i=1}^s b_i a_{ij}$ a (1.1), y donde se

ha tenido en cuenta que $\sum_{i=1}^s b_i = 1$ (condición necesaria y suficiente para la consistencia de un método Runge Kutta). Notamos que \mathbf{k}_i no es necesaria, y que bastaría renombrar \mathbf{k}'_i como \mathbf{k}_i .

El formato (1.2)–(1.4) correspondiente a un método Runge-Kutta-Nyström con coeficientes γ , \mathbf{b} , β y α arbitrarios es, por tanto, más general que el que se obtiene al aplicar un método Runge-Kutta al sistema de primer orden asociado (1.5), lo que justifica el estudio de los métodos Runge-Kutta-Nyström.

1.2. Condiciones de orden

En esta sección obtendremos condiciones sobre los coeficientes de un método Runge-Kutta-Nyström aplicado al sistema (1.1) que aseguran que el error local, visto como función de h es de un orden dado. Obtendremos dichas condiciones comparando los desarrollos de Taylor en t_0 de la solución exacta de (1.1) y de la solución numérica calculada mediante (1.2)–(1.4) con $n = 0$.

Definición 1.2.1. *Un método Runge-Kutta-Nyström tiene orden p , si para todo problema de segundo orden (1.1) con f suficientemente regular se cumple que*

$$\mathbf{y}(t_0 + h) - \mathbf{y}_1 = O(h^{p+1}), \quad \mathbf{v}(t_0 + h) - \mathbf{v}_1 = O(h^{p+1}), \quad (1.12)$$

siendo $\mathbf{y}(t_0 + h)$, $\mathbf{v}(t_0 + h)$ la solución exacta y su derivada en $t_0 + h$ e \mathbf{y}_1 , \mathbf{v}_1 la solución numérica tras un paso de longitud h .

Igual que sucede en el caso de los métodos Runge-Kutta, el hecho de que el error tras un paso (1.12) sea $O(h^{p+1})$ se traduce en que el error global (error cometido después de varios pasos) se comporta únicamente como $O(h^p)$ cuando h tiende a 0 (Ver [5, pág. 285]).

Si $\mathbf{y}(t)$ es la solución exacta de (1.1), notamos que $\mathbf{v}(t) = \mathbf{y}'(t)$, y denotamos por y^I y v^I la I -ésima componente de \mathbf{y} y \mathbf{v} respectivamente. Calcularemos primero las derivadas respecto de t de y^I . La segunda derivada está dada por (1.1) y es:

$$\frac{d^2}{dt^2}y^I = \frac{d}{dt}v^I = f^I(y^1, \dots, y^D). \quad (1.13)$$

Derivando repetidamente esta expresión obtenemos:

$$\frac{d^3}{dt^3}y^I = \sum_{J=1}^D \frac{\partial f^I}{\partial y^J} v^J, \quad (1.14)$$

$$\frac{d^4}{dt^4}y^I = \sum_{J,K=1}^D \frac{\partial^2 f^I}{\partial y^J \partial y^K} v^J v^K + \sum_{J=1}^D \frac{\partial f^I}{\partial y^J} f^J, \quad (1.15)$$

$$\begin{aligned}
\frac{d^5}{dt^5}y^I &= \sum_{J,K,L=1}^D \frac{\partial^3 f^I}{\partial y^J \partial y^K \partial y^L} v^J v^K v^L + \sum_{J,K=1}^D \frac{\partial^2 f^I}{\partial y^J \partial y^K} f^J v^K \\
&+ \sum_{J,K=1}^D \frac{\partial^2 f^I}{\partial y^J \partial y^K} v^J f^K + \sum_{J,K=1}^D \frac{\partial^2 f^I}{\partial y^J \partial y^K} f^J v^K \\
&+ \sum_{J,K=1}^D \frac{\partial f^I}{\partial y^J} \frac{\partial f^J}{\partial y^K} v^K.
\end{aligned} \tag{1.16}$$

Las derivadas sucesivas se pueden construir análogamente, aunque la continuación del proceso se vuelve cada vez más compleja (la siguiente derivada tendría 16 sumandos). Además podemos ver que algunos términos aparecen repetidos (como el segundo, el tercero y el cuarto sumandos de la derivada quinta). Por esta razón introducimos una herramienta gráfica, *los árboles especiales de Nyström o los SN-árboles con raíz*, que va a permitir una representación más compacta de las derivadas sucesivas de y^I .

Árboles y diferenciales elementales

Un árbol especial de Nyström con raíz (*SN-árbol con raíz*) de orden q es un grafo conexo y sin ciclos con q vértices (que pueden ser gruesos o finos) y un cierto número de lados conectando cada vértice (hijo) con un único vértice del nivel inferior (padre), de tal forma que en el primer nivel hay un único nodo que debe ser grueso (raíz del árbol), y se cumple además que cada nodo grueso solo puede tener hijos finos y que cada nodo fino puede tener a lo más un hijo que ha de ser grueso. A las aristas del grafo las llamaremos ramas. Al árbol con un único vértice (que ha de ser grueso) lo denotaremos por τ y cualquier *SN-árbol con raíz* t con q vértices, $q > 1$, se puede representar como se muestra en la Figura 1.1 y también con la notación $t = [r; t_1, t_2, \dots, t_m]$. Dicha notación indica que la raíz tiene r hijos finos finales (que no tienen hijos) y m nietos gruesos unidos a ella través de m vértices finos. Estos m vértices gruesos serán raíces de *SN-árboles con raíz* que denotaremos por t_1, \dots, t_m . Naturalmente, $r = 0$ o $m = 0$ son situaciones posibles. Si $r = 0$, escribiremos simplemente $[t_1, t_2, \dots, t_m]$ y al árbol τ con un único nodo lo denotaremos también por $[\emptyset]$.

Pasamos ahora a definir algunas de las funciones asociadas a los *SN-árboles con raíz* que vamos a necesitar más adelante.

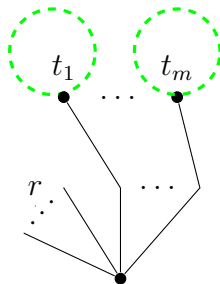


Figura 1.1: Representación del SN -árbol con raíz general $t=[r; t_1, t_2, \dots, t_m]$.

Definición 1.2.2. El orden de un SN -árbol con raíz, que denotaremos por $\rho(t)$, es el número de vértices del SN -árbol con raíz. Por tanto, $\rho(\tau) = 1$, y para un SN -árbol con raíz general $t = [r; t_1, \dots, t_m]$ como el de la Figura 1.1,

$$\rho(t) = r + 1 + \sum_{k=1}^m (\rho(t_k) + 1).$$

Definición 1.2.3. La diferencial elemental asociada a un SN -árbol con raíz t se denota por $F(t)(\mathbf{y})$ y, para un árbol general $t = [r; t_1, \dots, t_m]$ como el de la Figura 1.1, se define del siguiente modo:

$$F^I(t)(\mathbf{y}) = \sum_{J_1, \dots, J_r, K_1, \dots, K_m} \frac{\partial^{r+m} f^I(\mathbf{y})}{\partial y^{J_1} \dots \partial y^{J_r} \partial y^{K_1} \dots \partial y^{K_m}} v^{J_1} \dots v^{J_r} \cdot F^{K_1}(t_1)(\mathbf{y}) \dots F^{K_m}(t_m)(\mathbf{y}),$$

teniendo en cuenta que $F^I(\tau)(\mathbf{y}) = f^I(\mathbf{y})$, donde el superíndice I denota la I -ésima componente.

En el sumatorio anterior se puede identificar cada término v^{J_i} con cada hijo fino final de la raíz y cada término $F^{K_i}(t_i)(\mathbf{y})$ con cada uno de los SN -árboles con raíz t_i , que tienen como raíz a los nietos gruesos de la raíz de t . Observamos, por tanto, que el sumatorio está extendido a los hijos finos finales de la raíz y a sus nietos gruesos.

Como los subárboles t_1, \dots, t_m tienen menos nodos que el árbol original t , esta definición permite calcular recursivamente las diferenciales elementales de todos los árboles de un orden dado.

Ahora que hemos definido las diferenciales elementales asociadas a los SN -árboles con raíz estamos en condiciones de poder obtener las sucesivas derivadas respecto del tiempo de la solución $\mathbf{y}(t)$ como combinación lineal

de estas diferenciales elementales. Para ello, vamos a derivar, en primer lugar, las diferenciales elementales con la ayuda de los árboles asociados. Usaremos la regla de Leibniz para concluir que la derivación respecto del tiempo de una diferencial elemental da lugar a una suma de diferenciales elementales de SN -árboles con raíz con un nodo más.

Dada la diferencial elemental

$$F^I(t)(\mathbf{y}) = \sum_{J_1, \dots, J_r, K_1, \dots, K_m} \frac{\partial^{r+m} f^I(\mathbf{y})}{\partial y^{J_1} \dots \partial y^{J_r} \partial y^{K_1} \dots \partial y^{K_m}} v^{J_1} \dots v^{J_r} \cdot F^{K_1}(t_1)(\mathbf{y}) \dots F^{K_m}(t_m)(\mathbf{y}),$$

asociada a un SN -árbol con raíz general $t = [r; t_1, \dots, t_m]$ como el de la Figura 1.1, podemos derivar siguiendo las siguientes reglas:

- En el caso de derivar en el término $\frac{\partial^{r+m} f^I(\mathbf{y})}{\partial y^{J_1} \dots \partial y^{J_r} \partial y^{K_1} \dots \partial y^{K_m}}$ (asociado a la raíz de t) con respecto al tiempo, se obtiene

$$\sum_{J_1, \dots, J_r, J_{r+1}, K_1, \dots, K_m} \frac{\partial^{r+m+1} f^I}{\partial y^{J_1} \dots \partial y^{J_r} \partial y^{J_{r+1}} \partial y^{K_1} \dots \partial y^{K_m}} v^{J_1} \dots v^{J_r} v^{J_{r+1}} \cdot F^{K_1}(t_1)(\mathbf{y}) \dots F^{K_m}(t_m)(\mathbf{y}),$$

que se puede interpretar como la diferencial elemental asociada al SN -árbol con raíz $t' = [r+1; t_1, \dots, t_m]$ que se obtiene a partir de t añadiendo un nuevo hijo fino final a la raíz de t (ver Figura 1.2).

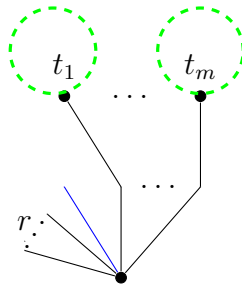


Figura 1.2: Representación del SN -árbol con raíz $t' = [r+1; t_1, \dots, t_m]$.

- También podemos derivar en un término v^{J_i} . Sin pérdida de generalidad derivamos en el término v^{J_1} con respecto al tiempo, y se obtiene

el término f^{J_1} . Renombrando el índice J_1 como K_{m+1} y teniendo en cuenta que $f^{K_{m+1}}(\mathbf{y}) = F^{K_{m+1}}(\tau)(\mathbf{y})$ obtenemos

$$\sum_{J_2, \dots, J_r, K_1, \dots, K_m, K_{m+1}} \frac{\partial^{r+m} f^I(\mathbf{y})}{\partial y^{J_2} \dots \partial y^{J_r} \partial y^{K_1} \dots \partial y^{K_m} \partial y^{K_{m+1}}} v^{J_2} \dots v^{J_r} \cdot F^{K_{m+1}}(\tau)(\mathbf{y}) F^{K_1}(t_1)(\mathbf{y}) \dots F^{K_m}(t_m)(\mathbf{y}),$$

que se puede interpretar como la diferencial elemental asociada al SN -árbol con raíz $t'' = [r - 1; [\emptyset], t_1, \dots, t_m]$ que se obtiene a partir de t añadiendo un nuevo hijo grueso a uno de los hijos finos finales de la raíz (ver Figura 1.3).

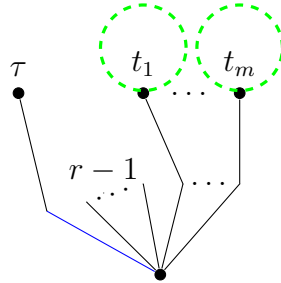


Figura 1.3: Representación del SN -árbol con raíz $t'' = [r - 1; [\emptyset], t_1, \dots, t_m]$.

- Por último, podemos derivar en un término $F^{K_i}(t_i)(\mathbf{y})$ con respecto al tiempo. Por el carácter recursivo de la definición de diferencial elemental podemos calcular del mismo modo expuesto anteriormente la derivada de $F^{K_i}(t_i)(\mathbf{y})$ puesto que el orden de t_i siempre es menor que el orden de t , se obtendrán diferenciales elementales asociadas a SN -árboles con raíz $t''' = [r; t_1, \dots, \tilde{t}_i, \dots, t_m]$ que se obtienen de t reemplazando t_i por SN -árboles con raíz \tilde{t}_i construidos a partir de t_i de manera análoga a como se han obtenido t' , t'' y t''' a partir de t .

Teniendo en cuenta que cada derivación aumenta el orden de los SN -árboles con raíz en una unidad y que la definición de diferencial elemental permite calcular recursivamente las diferenciales elementales de todos los árboles de un orden dado, podemos escribir las derivadas de la solución exacta de la siguiente forma

Teorema 1.2.1. *La solución exacta de (1.1) satisface:*

$$\mathbf{y}^{(q)} = \sum_{t \in SNT_{q-1}} \alpha(t) F(t)(\mathbf{y}) = \sum_{t \in LSNT_{q-1}} F(t)(\mathbf{y}) \quad (1.17)$$

donde SNT_{q-1} es el conjunto de los SN -árboles con raíz de orden $q - 1$, $LSNT_{q-1}$ es el conjunto de los SN -árboles etiquetados con raíz de orden $q - 1$ y $\alpha(t)$ son coeficientes enteros no negativos.

Demostración. Probaremos el resultado por inducción sobre $q \geq 2$. Para $q = 2$, se cumple el resultado porque

$$\frac{d^2}{dt^2}y^I = f^I(\mathbf{y}) = F^I(\tau)(\mathbf{y}).$$

Suponemos ahora que se cumple $\mathbf{y}^{(q)} = \sum_{t \in SNT_{q-1}} \alpha(t)F(t)(\mathbf{y})$, con $\alpha(t)$ entero no negativo, y probaremos el resultado para $q + 1$.

Cada uno de los términos que aparece en el sumatorio anterior está asociado a un SN -árbol con raíz t general como el de la Figura 1.1 con $q - 1$ nodos. Acabamos de ver que la derivada de la diferencial elemental asociada a un SN -árbol con raíz t es combinación lineal, con coeficientes enteros no negativos, de diferenciales elementales $F(\tilde{t})(\mathbf{y})$, donde $\tilde{t} \in SNT_q$ (tiene q nodos) se obtiene a partir de t , de la misma forma que se han construido t', t'' y t''' en el razonamiento previo al teorema. Se concluye entonces que

$$\mathbf{y}^{(q+1)} = \sum_{\tilde{t} \in SNT_q} \alpha(\tilde{t})F(\tilde{t})(\mathbf{y}),$$

donde $\alpha(\tilde{t})$ son coeficientes enteros no negativos. De hecho, aunque no lo probaremos en este trabajo, $\alpha(t)$ es el número de etiquetados monótonos del SN -árbol con raíz t (Ver [5, pág. 288]). Esto nos permite establecer la segunda igualdad de (1.17) en la que el sumatorio está extendido a todos los SN -árboles etiquetados con raíz. \square

Trataremos ahora de obtener una expresión similar a la del teorema anterior, pero, en este caso, para la solución numérica calculada mediante (1.2)–(1.4). Para ello, escribimos $\mathbf{k}_i = \mathbf{f}(\mathbf{g}_i)$ y reescribimos las ecuaciones (1.2)–(1.4) como

$$\mathbf{g}_i = \mathbf{y}_0 + h\gamma_i \mathbf{v}_0 + h^2 \sum_{j=1}^s \alpha_{ij} \mathbf{f}(\mathbf{g}_j), \quad (1.18)$$

$$\mathbf{y}_1 = \mathbf{y}_0 + h\mathbf{v}_0 + h^2 \sum_{i=1}^s \beta_i \mathbf{f}(\mathbf{g}_i), \quad (1.19)$$

$$\mathbf{v}_1 = \mathbf{v}_0 + h \sum_{i=1}^s b_i \mathbf{f}(\mathbf{g}_i), \quad (1.20)$$

Derivaremos en primer lugar la expresión \mathbf{g}_i respecto de h y dicha derivada será evaluada en $h = 0$. Como tenemos que derivar expresiones del tipo $h^2\varphi(h)$ y $h\varphi(h)$, usaremos la fórmula de Leibniz para la derivada q -ésima del producto, que aplicándola a nuestro caso nos da

$$(h\varphi(h))^{(q)}|_{h=0} = q(\varphi(h))^{(q-1)}|_{h=0}, \quad (1.21)$$

$$(h^2\varphi(h))^{(q)}|_{h=0} = q(q-1)(\varphi(h))^{(q-2)}|_{h=0}. \quad (1.22)$$

Calcularemos ahora las derivadas de g_i^I en $h = 0$ (el superíndice I denota la I -ésima componente del vector)

$$\begin{aligned} \frac{d}{dh}(g_i^I)|_{h=0} &= \gamma_i \mathbf{v}_0^I, \\ \frac{d^2}{dh^2}(g_i^I)|_{h=0} &= 2 \sum_{j=1}^s \alpha_{ij} f^I(\mathbf{g}_j)|_{h=0} = 2 \sum_{j=1}^s \alpha_{ij} f^I(\mathbf{y}_0), \\ \frac{d^3}{dh^3}(g_i^I)|_{h=0} &= 6 \sum_{j=1}^s \alpha_{ij} \frac{d}{dh}(f^I(\mathbf{g}_j))|_{h=0} = 6 \sum_{j=1}^s \alpha_{ij} \sum_{J=1}^D \frac{\partial f^I}{\partial y^J}(\mathbf{y}_0) \gamma_j \mathbf{v}_0^J \\ &= 6 \left(\sum_{j=1}^s \alpha_{ij} \gamma_j \right) \left(\sum_{J=1}^D \frac{\partial f^I}{\partial y^J}(\mathbf{y}_0) \mathbf{v}_0^J \right). \end{aligned}$$

Ahora bien, nos damos cuenta de que a partir de esta derivada, las derivadas sucesivas son más costosas de calcular y por ello, nos vemos motivados a encontrar una forma general para calcular la derivada q -ésima de \mathbf{g}_i . Por la similitud que \mathbf{g}_i guarda con \mathbf{y}_1 y \mathbf{v}_1 también encontraremos una forma general para calcular las derivadas de éstas últimas, pero para ello necesitamos definir dos nuevas funciones asociadas a los SN -árboles con raíz y enunciar un resultado que se va a utilizar más adelante.

Definición 1.2.4. *El peso elemental asociado a la etapa i -ésima del método y a un SN -árbol con raíz general $t = [r; t_1, \dots, t_m]$ como el de la Figura 1.1, se denota por $\Phi_i(t)$ y se calcula como*

$$\Phi_i(t) = \gamma_i^r \cdot \sum_{k_1, \dots, k_m=1}^s \alpha_{ik_1} \Phi_{k_1}(t_1) \cdots \alpha_{ik_m} \Phi_{k_m}(t_m),$$

teniendo en cuenta que $\Phi_i(\tau) = 1$ (τ denota el único SN -árbol con raíz con un nodo), y donde k_1, \dots, k_m son índices asociados a las raíces de los árboles t_1, \dots, t_m y r es el número de hijos finos finales de la raíz.

Definición 1.2.5. La función densidad asociada a un SN-árbol con raíz general $t = [r; t_1, \dots, t_m]$ como el de la Figura 1.1, que denotaremos por $\gamma(t)$, se define mediante

$$\gamma(t) = \rho(t) \cdot \prod_{l=1}^m ((\rho(t_l) + 1) \cdot \gamma(t_l)),$$

partiendo de $\gamma(\tau) = 1$, donde τ denota el único SN-árbol con raíz con un nodo y m es el número de nietos gruesos de la raíz de t .

Lema 1.2.1. (Fórmula de Faà di Bruno) [5, pág. 150] Para $q \geq 1$ se tiene que

$$(f^I(\mathbf{g}))^{(q-1)} = \sum_{u \in LS_q} \sum_{K_1, \dots, K_m} \frac{\partial^m f^I(\mathbf{g})}{\partial y^{K_1} \dots \partial y^{K_m}} (g^{K_1})^{(\delta_1)} \dots (g^{K_m})^{(\delta_m)}$$

donde para cada $u \in LS_q$, m es el número de ramas que parten de la raíz y $\delta_1, \dots, \delta_m$ denotan el número de nodos de cada una de esas ramas con lo que $q = 1 + \delta_1 + \dots + \delta_m$. (LS_q representa a los árboles con q nodos sin ramificaciones excepto en la raíz).

Utilizando las funciones asociadas a los SN-árboles con raíz y el Lema 1.2.1 estamos ahora en condiciones de enunciar el siguiente teorema que ofrece una expresión para la solución numérica similar a la del Teorema 1.2.1 para la solución exacta.

Teorema 1.2.2. Los vectores \mathbf{g}_i en (1.18) satisfacen

$$\frac{d^{q+1}}{dh^{q+1}}(g_i^I)|_{h=0} = (q+1) \sum_{t \in LSNT_q} \gamma(t) \sum_{j=1}^s \alpha_{ij} \Phi_j(t) F^I(t)(\mathbf{y}_0). \quad (1.23)$$

Demostración. Demostraremos el resultado aplicando un proceso de inducción sobre $q \geq 1$.

Si $q = 1$, tenemos que $\frac{d^2}{dh^2}(g_i^I)|_{h=0} = 2 \sum_{j=1}^s \alpha_{ij} f^I(\mathbf{y}_0)$. Teniendo en cuenta que $\gamma(\tau) = 1$, $\Phi_j(\tau) = 1$ y que $F^I(\tau)(\mathbf{y}) = f^I(\mathbf{y})$, se cumple el resultado

$$\frac{d^2}{dh^2}(g_i^I)|_{h=0} = 2\gamma(\tau) \sum_{j=1}^s \alpha_{ij} \Phi_j(\tau) F^I(\tau)(\mathbf{y}_0).$$

Suponemos ahora que se cumple

$$\frac{d^q}{dh^q}(g_i^I)|_{h=0} = q \sum_{t \in LSNT_{q-1}} \gamma(t) \sum_{j=1}^s \alpha_{ij} \Phi_j(t) F^I(t)(\mathbf{y}_0),$$

y probaremos el resultado para $q + 1$.

Vamos a calcular $\frac{d^{q+1}}{dh^{q+1}}(g_i^I)|_{h=0}$. Usaremos para ello la fórmula (1.22) y la fórmula de Faà di Bruno

$$\begin{aligned} \frac{d^{q+1}}{dh^{q+1}}(g_i^I)|_{h=0} &= (q+1)q \sum_{j=1}^s \alpha_{ij} \frac{d^{q-1}}{dh^{q-1}}(f^I(\mathbf{g}_j))|_{h=0} \\ &= (q+1) \sum_{j=1}^s \alpha_{ij} q \cdot \\ &\quad \cdot \sum_{LS_q} \sum_{J_1, \dots, J_r, K_1, \dots, K_m} \left(\frac{\partial^{r+m} f^I(\mathbf{g}_j)}{\partial y^{J_1} \dots \partial y^{J_r} \partial y^{K_1} \dots \partial y^{K_m}} \right) \Big|_{h=0} \cdot \\ &\quad \cdot ((g_j^{J_1})' \dots (g_j^{J_r})' (g_j^{K_1})^{(\delta_1)} \dots (g_j^{K_m})^{(\delta_m)}) \Big|_{h=0} \end{aligned}$$

Por (1.18), $(g_j^{J_i})'|_{h=0} = \gamma_j v_0^{J_i}$. Además K_i hace referencia a la raíz del SN -árbol con raíz t_i para el que tenemos que $\rho(t_i) < q$. Por la naturaleza recursiva de las definiciones y evaluando en $h = 0$ obtenemos que

$$\begin{aligned} \frac{d^{q+1}}{dh^{q+1}}(g_i^I)|_{h=0} &= (q+1) \sum_{j=1}^s \alpha_{ij} q \\ &\quad \cdot \sum_{LS_q} \sum_{J_1, \dots, J_r, K_1, \dots, K_m} \gamma_j^r \frac{\partial^{r+m} f^I(\mathbf{y}_0)}{\partial y^{J_1} \dots \partial y^{J_r} \partial y^{K_1} \dots \partial y^{K_m}} \\ &\quad \cdot \delta_1 \sum_{t_1 \in LSNT_{\delta_1-1}} \gamma(t_1) \sum_{k_1=1}^s \alpha_{jk_1} \Phi_{k_1}(t_1) F^{K_1}(t_1)(\mathbf{y}_0) \dots \\ &\quad \dots \delta_m \sum_{t_m \in LSNT_{\delta_m-1}} \gamma(t_m) \sum_{k_m=1}^s \alpha_{jk_m} \Phi_{k_m}(t_m) F^{K_m}(t_m)(\mathbf{y}_0). \end{aligned}$$

Dado $u \in LS_q$, $t_1 \in LSNT_{\delta_1-1}$, \dots , $t_m \in LSNT_{\delta_m-1}$ se puede considerar

$t \in LSNT_q$ como el de la Figura 1.1 para el que

$$\begin{aligned}\gamma(t) &= q \cdot \delta_1 \cdot \gamma(t_1) \cdots \delta_m \cdot \gamma(t_m), \\ \Phi_j(t) &= \gamma_j^r \sum_{k_1, \dots, k_m=1}^s \alpha_{jk_1} \Phi_{k_1}(t_1) \cdots \alpha_{jk_m} \Phi_{k_m}(t_m),\end{aligned}$$

y se concluye entonces que

$$\frac{d^{q+1}}{dh^{q+1}}(g_i^I)|_{h=0} = (q+1) \sum_{t \in LSNT_q} \gamma(t) \sum_{j=1}^s \alpha_{ij} \Phi_j(t) F^I(t)(\mathbf{y}_0).$$

□

Como consecuencia de la semejanza entre \mathbf{g}_i e \mathbf{y}_1 y \mathbf{v}_1 obtenemos el siguiente teorema

Teorema 1.2.3. *La solución numérica \mathbf{y}_1 y \mathbf{v}_1 dada por (1.18)–(1.20) satisface*

$$\frac{d^q}{dh^q}(\mathbf{y}_1)|_{h=0} = q \sum_{t \in LSNT_{q-1}} \gamma(t) \sum_{i=1}^s \beta_i \Phi_i(t) F(t)(\mathbf{y}_0), \quad (1.24)$$

$$\frac{d^{q-1}}{dh^{q-1}}(\mathbf{v}_1)|_{h=0} = \sum_{t \in LSNT_{q-1}} \gamma(t) \sum_{i=1}^s b_i \Phi_i(t) F(t)(\mathbf{y}_0). \quad (1.25)$$

Demostración. Para la primera igualdad basta comparar

$$\mathbf{g}_i = \mathbf{y}_0 + \gamma_i h \mathbf{v}_0 + h^2 \sum_{j=1}^s \alpha_{ij} \mathbf{f}(\mathbf{g}_j),$$

con

$$\mathbf{y}_1 = \mathbf{y}_0 + h \mathbf{v}_0 + h^2 \sum_{i=1}^s \beta_i \mathbf{f}(\mathbf{g}_i),$$

para darnos cuenta que es suficiente sustituir en (1.23) α_{ij} por β_i y el sumatorio en j por el sumatorio en i para obtener el resultado buscado (1.24).

Para el segundo caso, aplicaremos la fórmula (1.21) para derivar \mathbf{v}_1

$$\frac{d^q}{dh^q}(\mathbf{v}_1)|_{h=0} = q \sum_{i=1}^s b_i \frac{d^{q-1}}{dh^{q-1}}(f^I((\mathbf{g}_j))|_{h=0},$$

que comparándolo con la derivada $(q + 1)$ -ésima de g_i^I después de aplicar la fórmula de derivación (1.22)

$$\frac{d^{q+1}}{dh^{q+1}}(g_i^I)|_{h=0} = (q + 1)q \sum_{j=1}^s \alpha_{ij} \frac{d^{q-1}}{dh^{q-1}}(f^I((\mathbf{g}_j))|_{h=0}$$

nos permite, como en el caso anterior, sustituir en (1.23) α_{ij} por b_i , el sumatorio en j por el sumatorio en i y dividir por $q + 1$ para obtener el resultado buscado (1.25). \square

Estamos ya en disposición de obtener finalmente las condiciones de orden de los métodos Runge-Kutta-Nyström igualando los desarrollos de Taylor de la solución exacta y numérica obtenidos en esta sección.

Teorema 1.2.4. *Un método Runge-Kutta-Nyström (1.2)–(1.4) para (1.1) tiene orden $q \geq 2$ si para cada SN-árbol con raíz t con $\rho(t) \leq q - 1$ se verifica*

$$\sum_{i=1}^s \beta_i \Phi_i(t) = \frac{1}{(1 + \rho(t))\gamma(t)}, \quad (1.26)$$

y para cada SN-árbol con raíz t con $\rho(t) \leq q$ se verifica

$$\sum_{i=1}^s b_i \Phi_i(t) = \frac{1}{\gamma(t)}. \quad (1.27)$$

Nota 1.1. *Para tener orden 1 es suficiente pedir que $\sum_{i=1}^s b_i = 1$.*

Nota 1.2. *Aunque no se demostrará en este trabajo, se cumple que las condiciones (1.26) y (1.27) son también necesarias (Ver [1, Pag. 32-37]).*

Condiciones de orden hasta orden 5

Teniendo en cuenta (1.28) y (1.29), y apoyándonos en las Tablas 1.1 y 1.2, donde se han representado los SN-árboles con raíz de orden menor o igual que 5, escribimos a continuación las correspondientes condiciones de orden.

- **Orden 1:** $\sum_{i=1}^s b_i = 1$.

- **Orden 2:** Además de la condición anterior debe cumplirse

$$\sum_{i=1}^s b_i \gamma_i = \frac{1}{2}, \quad \sum_{i=1}^s \beta_i = \frac{1}{2}.$$

- **Orden 3:** Además de las 3 condiciones anteriores debe cumplirse

$$\sum_{i=1}^s b_i \gamma_i^2 = \frac{1}{3}, \quad \sum_{i,j=1}^s b_i \alpha_{ij} = \frac{1}{6}, \quad \sum_{i=1}^s \beta_i \gamma_i = \frac{1}{6}.$$

- **Orden 4:** Además de las 6 condiciones anteriores debe cumplirse

$$\sum_{i=1}^s b_i \gamma_i^3 = \frac{1}{4}, \quad \sum_{i,j=1}^s b_i \gamma_i \alpha_{ij} = \frac{1}{8}, \quad \sum_{i,j=1}^s b_i \alpha_{ij} \gamma_j = \frac{1}{24},$$

$$\sum_{i=1}^s \beta_i \gamma_i^2 = \frac{1}{12}, \quad \sum_{i,j=1}^s \beta_i \alpha_{ij} = \frac{1}{24}.$$

- **Orden 5:** Además de las 11 condiciones anteriores debe cumplirse

$$\sum_{i=1}^s b_i \gamma_i^4 = \frac{1}{5}, \quad \sum_{i,j=1}^s b_i \gamma_i^2 \alpha_{ij} = \frac{1}{10}, \quad \sum_{i,j=1}^s b_i \alpha_{ij} \alpha_{ik} = \frac{1}{20},$$

$$\sum_{i,j=1}^s b_i \gamma_i \alpha_{ij} \gamma_j = \frac{1}{30}, \quad \sum_{i,j=1}^s b_i \alpha_{ij} \gamma_j^2 = \frac{1}{60}, \quad \sum_{i,j,k=1}^s b_i \alpha_{ij} \alpha_{jk} = \frac{1}{120},$$

$$\sum_{i=1}^s \beta_i \gamma_i^3 = \frac{1}{20}, \quad \sum_{i,j=1}^s \beta_i \gamma_i \alpha_{ij} = \frac{1}{40}, \quad \sum_{i,j=1}^s \beta_i \alpha_{ij} \gamma_j = \frac{1}{120}.$$






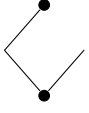

	t	$\gamma(t)$	$\mathbf{F}(t)(\mathbf{y})$	$\Phi_i(t)$
t_1	 [\emptyset]	1	f^I	1
t_2	 [1]	2	$\sum_{J=1}^D \frac{\partial f^I}{\partial y^J} v^J$	γ_i
t_3	 [2]	3	$\sum_{J,K=1}^D \frac{\partial^2 f^I}{\partial y^J \partial y^K} v^J v^K$	γ_i^2
t_4	 [[\emptyset]]	6	$\sum_{J=1}^D \frac{\partial f^I}{\partial y^J} f^J$	$\sum_{j=1}^s \alpha_{ij}$
t_5	 [3]	4	$\sum_{J,K,L=1}^D \frac{\partial^3 f^I}{\partial y^J \partial y^K \partial y^L} v^J v^K v^L$	γ_i^3
t_6	 [1; [\emptyset]]	8	$\sum_{J,K=1}^D \frac{\partial^2 f^I}{\partial y^J \partial y^K} f^J v^K$	$\gamma_i \sum_{j=1}^s \alpha_{ij}$
t_7	 [[1]]	24	$\sum_{J,K=1}^D \frac{\partial f^I}{\partial y^J} \frac{\partial f^J}{\partial y^K} v^K$	$\sum_{j=1}^s \alpha_{ij} \gamma_j$

Tabla 1.1: SN -árboles con raíz y diferenciales elementales hasta orden 4







	t	$\gamma(t)$	$\mathbf{F}(t)(\mathbf{y})$	$\Phi_i(t)$
t_8	 [4]	5	$\sum_{J,K,L,M=1}^D \frac{\partial^3 f^I}{\partial y^J \partial y^K \partial y^L \partial y^M} v^J v^K v^L v^M$	γ_i^4
t_9	 [2; [∅]]	10	$\sum_{J,K,L=1}^D \frac{\partial^2 f^I}{\partial y^J \partial y^K \partial y^L} f^J v^K v^L$	$\gamma_i^2 \sum_{j=1}^s \alpha_{ij}$
t_{10}	 [[∅], [∅]]	20	$\sum_{J,K=1}^D \frac{\partial^2 f^I}{\partial y^J \partial y^K} f^J f^K$	$\sum_{j,k=1}^s \alpha_{ij} \alpha_{ik}$
t_{11}	 [1; [1]]	30	$\sum_{J,K,L=1}^D \frac{\partial^2 f^I}{\partial y^J \partial y^K} \frac{\partial f^J}{\partial y^L} v^L v^K$	$\gamma_i \sum_{j=1}^s \alpha_{ij} \gamma_j$
t_{12}	 [[2]]	60	$\sum_{J,K,L=1}^D \frac{\partial f^I}{\partial y^J} \frac{\partial^2 f^J}{\partial y^K \partial y^L} v^K v^L$	$\sum_{j=1}^s \alpha_{ij} \gamma_j^2$
t_{13}	 [[[∅]]]	120	$\sum_{J,K=1}^D \frac{\partial f^I}{\partial y^J} \frac{\partial f^J}{\partial y^K} f^K$	$\sum_{j,k=1}^s \alpha_{ij} \alpha_{jk}$

Tabla 1.2: SN -árboles con raíz y diferenciales elementales de orden 5

Enunciaremos ahora algunos resultados que son útiles a la hora de reducir el número de condiciones de orden que hay que imponer en la construcción de métodos Runge-Kutta-Nyström cuando sus coeficientes satisfacen ciertas relaciones, llamadas habitualmente *condiciones simplificadoras*.

Lema 1.2.2. *Si los coeficientes del método RKN satisfacen*

$$\beta_i = b_i(1 - \gamma_i) \quad i = 1, \dots, s, \quad (1.28)$$

y para cada *SN*-árbol con raíz t con $\rho(t) \leq q$ se cumple (1.27), entonces se cumple (1.26) para cada *SN*-árbol con raíz t con $\rho(t) \leq q - 1$.

Demostración. Sea t un *SN*-árbol de orden $\rho(t) \leq q - 1$ y sea \tilde{t} el *SN*-árbol con raíz de orden $\rho(t) + 1$ que se obtiene añadiendo un hijo fino final a la raíz de t . Por la construcción de \tilde{t} y por la Definición 1.2.4 tenemos que

$$\Phi_i(\tilde{t}) = \gamma_i \Phi_i(t),$$

y por la Definición 1.2.5 tenemos que

$$\gamma(\tilde{t}) = \rho(\tilde{t}) \frac{\gamma(t)}{\rho(t)} = (\rho(t) + 1) \frac{\gamma(t)}{\rho(t)}.$$

Teniendo en cuenta estas dos igualdades y la hipótesis (1.28), obtenemos

$$\begin{aligned} \sum_{i=1}^s \beta_i \Phi_i(t) &= \sum_{i=1}^s b_i(1 - \gamma_i) \Phi_i(t) = \sum_{i=1}^s b_i \Phi_i(t) - \sum_{i=1}^s b_i \gamma_i \Phi_i(t) \\ &= \sum_{i=1}^s b_i \Phi_i(t) - \sum_{i=1}^s b_i \Phi_i(\tilde{t}) = \frac{1}{\gamma(t)} - \frac{1}{\gamma(\tilde{t})} \\ &= \frac{1}{\gamma(t)} - \frac{1}{(\rho(t) + 1) \frac{\gamma(t)}{\rho(t)}} \\ &= \frac{(\rho(t) + 1) - \rho(t)}{(\rho(t) + 1)\gamma(t)} = \frac{1}{(1 + \rho(t))\gamma(t)}. \end{aligned}$$

En consecuencia, la condición de orden (1.26) asociada al árbol t se satisface automáticamente. En las igualdades anteriores hemos utilizado que tanto t como \tilde{t} son *SN*-árboles con raíz de orden $\leq q$ para los que se cumple, por tanto, (1.27). \square

Lema 1.2.3. Sean t y u dos SN-árboles con raíz de orden q , como los de la Figura 1.4, donde se supone que las partes que están dentro de la elipse deben ser idénticas en los dos árboles. Si se cumple

$$\sum_{j=1}^s \alpha_{ij} = \frac{\gamma_i^2}{2}, \quad i = 1, \dots, s, \quad (1.29)$$

entonces las condiciones de orden asociadas a t y u son equivalentes.

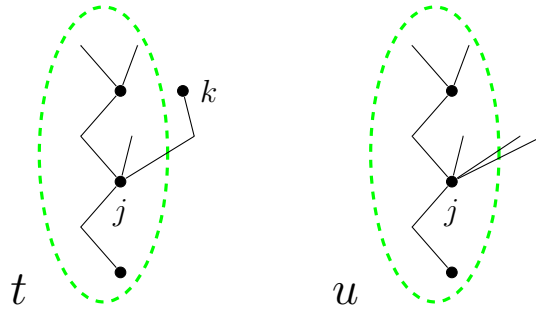


Figura 1.4: Representación gráfica de los árboles asociados al Lema 1.2.3 (t en la izquierda y u en la derecha).

Demostración. Sea \tilde{t} el SN-árbol con raíz de orden $q - 2$ que representa a la parte común de t y u . Por el carácter recurrente de la Definición 1.2.4 y por (1.29) obtenemos que

$$\Phi_i(t) = \Phi_i(\tilde{t}) \sum_{k=1}^s \alpha_{jk} = \Phi_i(\tilde{t}) \frac{\gamma_j^2}{2} \quad y \quad \Phi_i(u) = \Phi_i(\tilde{t}) \gamma_j^2,$$

siendo j el índice del vértice grueso que está dentro de la parte común de t y u en el que se produce la diferencia entre t y u y del que derivan los dos vértices que están fuera de esta parte común, y siendo k el vértice grueso final de t que está fuera de la parte común de t y u (Ver Figura 1.4). Tenemos entonces que

$$2\Phi_i(t) = \Phi_i(u).$$

Por otro lado, por el carácter recurrente de la Definición 1.2.5 tenemos que

$$\gamma(t) = \gamma(\tilde{t}) \cdot 2 \quad y \quad \gamma(u) = \gamma(\tilde{t}).$$

Esto implica

$$\frac{\gamma(t)}{2} = \gamma(u)$$

y, por tanto,

$$\Phi_i(t)\gamma(t) = \Phi_i(u)\gamma(u),$$

de donde deducimos que $\Phi_i(t) = \frac{1}{\gamma(t)}$ es equivalente a $\Phi_i(u) = \frac{1}{\gamma(u)}$ y basta considerar sólo una de dichas condiciones de orden. \square

Como consecuencia del Lema 1.2.3 las condiciones de orden de los árboles t_8 , t_9 y t_{10} de la Tabla 1.2 son equivalentes y lo mismo sucede con las condiciones de orden de los árboles t_{12} y t_{13} , pasando de 6 condiciones asociadas a SN -árboles con raíz de orden 5 a solamente 3.

Contando el número de condiciones de orden

Veamos ahora cuántas condiciones de orden son necesarias a medida que aumenta el orden del método [1, Pág 41]. Vamos a denotar por m_N el número de SN -árboles con raíz distintos de orden N , $N = 1, 2, \dots$. Es claro que $m_1 = 1$. Cada uno de estos m_N SN -árboles con raíz estará formado por la raíz y partiendo de ella habrá $j_1 + j_2 + \dots + j_k$ nodos finos de los que j_1 son nodos finos finales, de j_2 parten j_2 SN -árboles con raíz de 1 nodo, de j_3 parten j_3 SN -árboles con raíz de 2 nodos, \dots , de j_k parten j_k SN -árboles con raíz de $k - 1$ nodos de modo que $j_1 + 2j_2 + \dots + kj_k = N - 1$. Notamos ahora que para $k > 1$, los j_k SN -árboles de $k - 1$ nodos se pueden escoger de

$$\binom{m_{k-1} + j_k - 1}{j_k}$$

formas diferentes (hay una sola manera de elegir los j_2 SN -árboles con raíz con un nodo). Entonces la fórmula que nos permite calcular recursivamente m_N es

$$m_N = \sum_{j_1 + 2j_2 + \dots + kj_k = N-1} \binom{m_1 + j_2 - 1}{j_2} \dots \binom{m_{k-1} + j_k - 1}{j_k}, \quad (1.30)$$

junto con $m_1 = 1$. Podemos definir la función generatriz M de m_N como la serie formal de potencias

$$M(z) = \sum_{N=1}^{\infty} m_N z^N, \quad (1.31)$$

que a la vista del número de SN -árboles con raíz de las Tablas 1.1 y 1.2 tiene la forma

$$M(z) = z + z^2 + 2z^3 + 3z^4 + 6z^5 + \dots$$

Teniendo en cuenta [4, pág 363] que para $r, m \leq 1$

$$\frac{1}{(1-z^r)^m} = \sum_{j=0}^{\infty} \binom{m+j-1}{j} z^{rj}$$

y la relación (1.30) se concluye que

$$M(z) = \frac{z}{(1-z)(1-z^2)^{m_1} \cdots (1-z^k)^{m_{k-1}} \cdots}. \quad (1.32)$$

Tomando logaritmos en (1.32) se tiene

$$\ln \left(\frac{M(z)}{z} \right) = \ln \left(\frac{1}{1-z} \right) + m_1 \ln \left(\frac{1}{1-z^2} \right) + \cdots + m_{k-1} \ln \left(\frac{1}{1-z^k} \right) + \cdots,$$

esto es,

$$\ln \left(\frac{M(z)}{z} \right) = -\ln(1-z) - m_1 \ln(1-z^2) - \cdots - m_{k-1} \ln(1-z^k) - \cdots.$$

Como consecuencia del desarrollo en serie de la función logaritmo obtenemos

que $\ln(1-z^r) = -\sum_{n=1}^{\infty} \frac{z^{rn}}{n}$, de donde deducimos que

$$\begin{aligned} \ln \left(\frac{M(z)}{z} \right) &= \sum_{n=1}^{\infty} \frac{z^n}{n} + m_1 \sum_{n=1}^{\infty} \frac{z^{2n}}{n} + \cdots + m_{k-1} \sum_{n=1}^{\infty} \frac{z^{kn}}{n} + \cdots \\ &= \sum_{n=1}^{\infty} \frac{z^n}{n} (1 + m_1 z^n + m_2 z^{2n} + \cdots + m_{k-1} z^{(k-1)n} + \cdots) \\ &= \sum_{n=1}^{\infty} \frac{z^n}{n} (1 + M(z^n)). \end{aligned}$$

Tomando exponenciales obtenemos

$$M(z) = z \cdot \exp \left(z(1 + M(z)) + \cdots + \frac{z^k}{k} (1 + M(z^k)) + \cdots \right),$$

y por tanto, es posible escribir el producto infinito en (1.32) en términos de una exponencial, que permite también el cálculo recursivo de m_N .

En la Tabla 1.3 aparecen los valores de m_N para $N \leq 9$. Observamos que para obtener un método RKN general de orden 4 hay que imponer

N	1	2	3	4	5	6	7	8	9
m_N	1	1	2	3	6	10	20	36	72

Tabla 1.3: Valores de m_N para $N \leq 9$.

$(1 + 1 + 2 + 3) + (1 + 1 + 2) = 7 + 4 = 11$ condiciones de orden, mientras que para métodos RKN que satisfacen (1.28) es suficiente imponer 7 condiciones de orden (las asociadas a los SN -árboles con raíz de la Tabla 1.1). Por otro lado, si además de (1.28) se satisface (1.29) observamos que las condiciones de orden de t_3 y t_4 son equivalentes y lo mismo sucede con las condiciones de orden para t_5 y t_6 . Se concluye entonces que si se cumplen (1.28) y (1.29) es suficiente imponer 5 condiciones de orden para obtener métodos de orden 4. También observamos en la Tabla 1.3 que para obtener un método RKN general de orden 6 hay que imponer $(1 + 1 + 2 + 3 + 6 + 10) + (1 + 1 + 2 + 3 + 6) = 23 + 13 = 36$ condiciones de orden, mientras que para métodos RKN que satisfacen (1.28) es suficiente imponer 23 condiciones de orden. Aunque en la memoria no se han incluido los 10 SN -árboles con raíz de orden 6, se puede comprobar que si se cumplen (1.28) y (1.29) es suficiente imponer 5 condiciones de orden asociadas a SN -árboles con raíz con 6 nodos, que junto con las 3 condiciones para árboles de orden 5 y las 5 condiciones para árboles de orden menor o igual que 4 dan un total de 13 condiciones de orden que hay que imponer para obtener un método RKN de orden 6. Esto implica una reducción importante del número de condiciones de orden, pasando de 36 condiciones de orden para un método general de orden 6, a sólo 13 si el método cumple (1.28) y (1.29).

1.3. Estabilidad lineal

La estabilidad lineal de un método Runge-Kutta-Nyström es investigada mediante la ecuación test escalar

$$y'' = -\omega^2 y, \quad \omega > 0, \quad (1.33)$$

cuya solución exacta, $y(t) = A \cos(\omega t + \theta)$, con A y θ dependiendo de las condiciones iniciales, es oscilatoria [2]. La aplicación de un método Runge-Kutta-Nyström de s etapas con coeficientes γ , \mathbf{b} , $\boldsymbol{\beta}$, y α a la integración numérica de (1.33) da lugar a

$$y_{n+1} = y_n + h v_n - h^2 \omega^2 \boldsymbol{\beta}^T \mathbf{Y}_n, \quad (1.34)$$

$$v_{n+1} = v_n - h \omega^2 \mathbf{b}^T \mathbf{Y}_n, \quad (1.35)$$

con \mathbf{Y}_n el vector de etapas intermedias del método, que se puede escribir como

$$\mathbf{Y}_n = y_n \mathbf{e} + hv_n \boldsymbol{\gamma} - h^2 \omega^2 \alpha \mathbf{Y}_n,$$

donde \mathbf{e} es el vector de \mathbb{R}^s cuyas componentes son todas iguales a 1. Despejamos \mathbf{Y}_n de la ecuación anterior del siguiente modo

$$\mathbf{Y}_n = (I + h^2 \omega^2 \alpha)^{-1} (y_n \mathbf{e} + hv_n \boldsymbol{\gamma}),$$

y llamando $z = -h^2 \omega^2$ tenemos que

$$\mathbf{Y}_n = (I - z\alpha)^{-1} (y_n \mathbf{e} + hv_n \boldsymbol{\gamma}). \quad (1.36)$$

La matriz $(I - z\alpha)$ es siempre invertible en nuestro caso, pues estamos trabajando con métodos explícitos y por tanto, α es estrictamente triangular inferior.

Estamos ahora en condiciones de calcular la matriz de estabilidad del método, $M(z)$, que es la que satisface

$$\begin{bmatrix} y_{n+1} \\ hv_{n+1} \end{bmatrix} = M(z) \begin{bmatrix} y_n \\ hv_n \end{bmatrix}, \quad n = 0, 1, 2, \dots$$

Para hacer el cálculo de $M(z)$ llevaremos (1.36) a (1.34) para obtener

$$\begin{aligned} y_{n+1} &= y_n + hv_n - h^2 \omega^2 \boldsymbol{\beta}^T \mathbf{Y}_n \\ &= y_n + hv_n + z \boldsymbol{\beta}^T (I - z\alpha)^{-1} (y_n \mathbf{e} + hv_n \boldsymbol{\gamma}) \\ &= (1 + z \boldsymbol{\beta}^T (I - z\alpha)^{-1} \mathbf{e}) y_n + (1 + z \boldsymbol{\beta}^T (I - z\alpha)^{-1} \boldsymbol{\gamma}) hv_n. \end{aligned}$$

Del mismo modo, reemplazando \mathbf{Y}_n por (1.36) en (1.35) tenemos

$$\begin{aligned} hv_{n+1} &= h(v_n - h\omega^2 \mathbf{b}^T \mathbf{Y}_n) \\ &= hv_n - h^2 \omega^2 \mathbf{b}^T ((I - z\alpha)^{-1} (y_n \mathbf{e} + hv_n \boldsymbol{\gamma})) \\ &= hv_n + z \mathbf{b}^T ((I - z\alpha)^{-1} (y_n \mathbf{e} + hv_n \boldsymbol{\gamma})) \\ &= (z \mathbf{b}^T (I - z\alpha)^{-1} \mathbf{e}) y_n + (1 + z \mathbf{b}^T ((I - z\alpha)^{-1} \boldsymbol{\gamma})) hv_n. \end{aligned}$$

Por tanto,

$$M(z) = \begin{bmatrix} 1 + z \boldsymbol{\beta}^T (I - z\alpha)^{-1} \mathbf{e} & 1 + z \boldsymbol{\beta}^T (I - z\alpha)^{-1} \boldsymbol{\gamma} \\ z \mathbf{b}^T (I - z\alpha)^{-1} \mathbf{e} & 1 + z \mathbf{b}^T (I - z\alpha)^{-1} \boldsymbol{\gamma} \end{bmatrix}. \quad (1.37)$$

El efecto de amortiguación de $M(z)$ está caracterizado por su radio espectral, $\rho(M(z))$. Además, como $M(z)$ es una matriz 2×2 , los autovalores de $M(z)$ son las raíces de la ecuación característica

$$\mu^2 - S(z)\mu + P(z) = 0, \quad (1.38)$$

donde $S(z)$ es la traza de la matriz de estabilidad y $P(z)$ su determinante, es decir,

$$\begin{aligned} S(z) &= 2 + z(\boldsymbol{\beta}^T(I - z\alpha)^{-1}\mathbf{e} + \mathbf{b}^T(I - z\alpha)^{-1}\boldsymbol{\gamma}), \\ P(z) &= 1 + z(\boldsymbol{\beta}^T(I - z\alpha)^{-1}\mathbf{e} + \mathbf{b}^T(I - z\alpha)^{-1}(\boldsymbol{\gamma} - \mathbf{e}) + \\ &\quad z(\boldsymbol{\beta}^T(I - z\alpha)^{-1})(\mathbf{e}\mathbf{b}^T(I - z\alpha)^{-1}\boldsymbol{\gamma} - \boldsymbol{\gamma}\mathbf{b}^T(I - z\alpha)^{-1}\mathbf{e})). \end{aligned}$$

Definimos ahora algunas propiedades de estabilidad y periodicidad de $M(z)$.

Definición 1.3.1. *La región de estabilidad absoluta de un método Runge-Kutta-Nyström está definida por el conjunto*

$$\{z < 0 : \rho(M(z)) < 1\}. \quad (1.39)$$

Si la región de estabilidad absoluta es $(-\infty, 0)$ se dice que el método Runge-Kutta-Nyström es A-estable.

Como se puede ver en el Apéndice B, la región de estabilidad absoluta (1.39) se puede definir alternativamente reemplazando (1.39) por

$$\{z : z < 0, |P(z)| < 1, |S(z)| < P(z) + 1\}. \quad (1.40)$$

Los métodos A-estables amortiguan el error pero generan soluciones numéricas para (1.33) que espiralan hacia el origen (Ver Figura 1.8) en lugar de generar soluciones periódicas como la solución exacta de (1.33).

Hemos estudiado la región de estabilidad del método RKN4(3)4FM presentado en [3], y utilizado en los experimentos numéricos del Capítulo 2. En la Figura 1.5 se han representado los módulos de los dos autovalores, μ_1 y μ_2 , de $M(z)$ para una red equiespaciada de centésima en centésima de valores $z \in (-16, 0)$. Observamos que para $z > -9$ hay una única línea que corresponde a un par de autovalores complejos conjugados. Para $z < -9$ se observan dos autovalores reales distintos que se mantienen con módulo < 1 hasta llegar a -14.25 , luego obtenemos una región de estabilidad absoluta $(-14.25, 0)$ y por tanto, no es un método A-estable. En la Figura 1.6 se han representado los módulos de la traza y el determinante de $M(z)$, el discriminante de (1.38) y $P(z) + 1$ para los mismos valores de z considerados en la Figura 1.5. Observamos que para $z = 14.25$, deja de cumplirse $|S(z)| < P(z) + 1$, una de las condiciones necesarias según (1.40). También notamos que el discriminante de (1.38), $S(z)^2 - 4P(z)$, es negativo desde 0 hasta -9 , punto en el que los autovalores pasan de ser complejos conjugados a un par de autovalores reales distintos, concordando con lo que habíamos observado en la Figura 1.5.

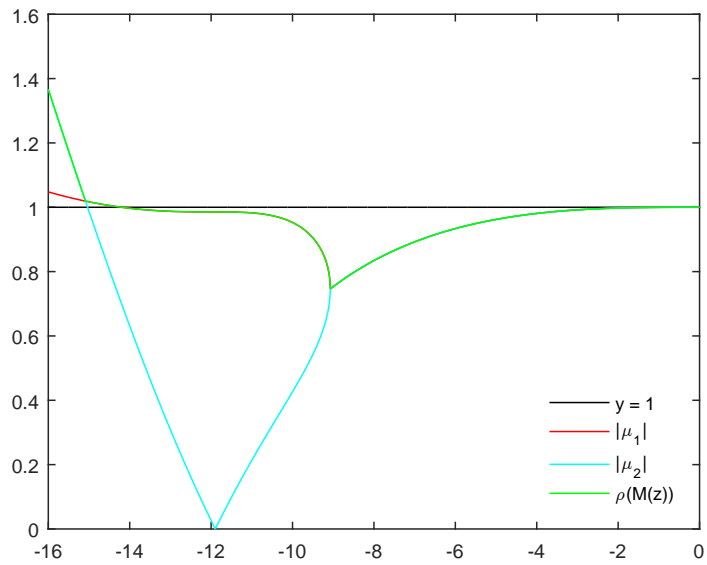


Figura 1.5: Región de estabilidad del método RKN4(3)4FM, utilizando (1.39)

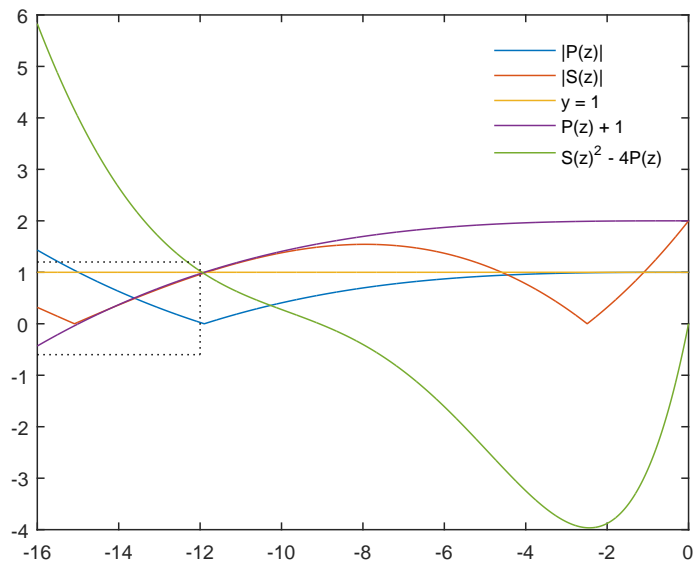


Figura 1.6: Región de estabilidad del método RKN4(3)4FM, utilizando (1.42)

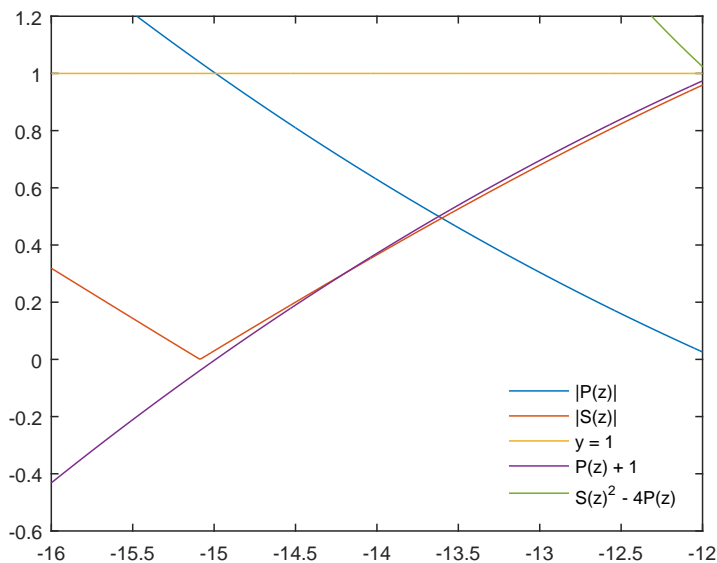


Figura 1.7: Detalle y ampliación de la Figura 1.6

La Figura 1.7 es una ampliación del rectángulo punteado de la Figura 1.6. En ella se aprecia con claridad que la condición $|S(z)| < P(z) + 1$ de (1.40) deja de cumplirse en el punto 14.25, aunque sí se cumple todavía la condición $|P(z)| < 1$. Por otro lado, en la Figura 1.8 hemos representado la solución numérica para la ecuación test (1.35) con $\omega = 10$ generada con longitud de paso $h = 0.2$ ($h^2\omega^2 = -4$). Se han numerado las 10 primeras aproximaciones calculadas y las 10 últimas, de modo que se pueda apreciar que efectivamente la solución numérica calculada con este método espirala hacia el origen.

Definición 1.3.2. *La región de periodicidad de un método Runge-Kutta-Nyström está definida por el conjunto*

$$\{z < 0 : \rho(M(z)) = 1, [S(z)]^2 - 4P(z) < 0\}. \quad (1.41)$$

Si la región de periodicidad es $(-\infty, 0)$ se dice que el método Runge-Kutta-Nyström es P-estable.

Para z en el dominio de periodicidad, los autovalores de $M(z)$ son un par de complejos conjugados de módulo unidad, por lo que la solución numérica es periódica tal y como lo es la solución exacta. Hemos estudiado la región de periodicidad del método RKN4(3)4FM y como se puede ver en la Figura 1.5 la región de periodicidad de este método es vacía.

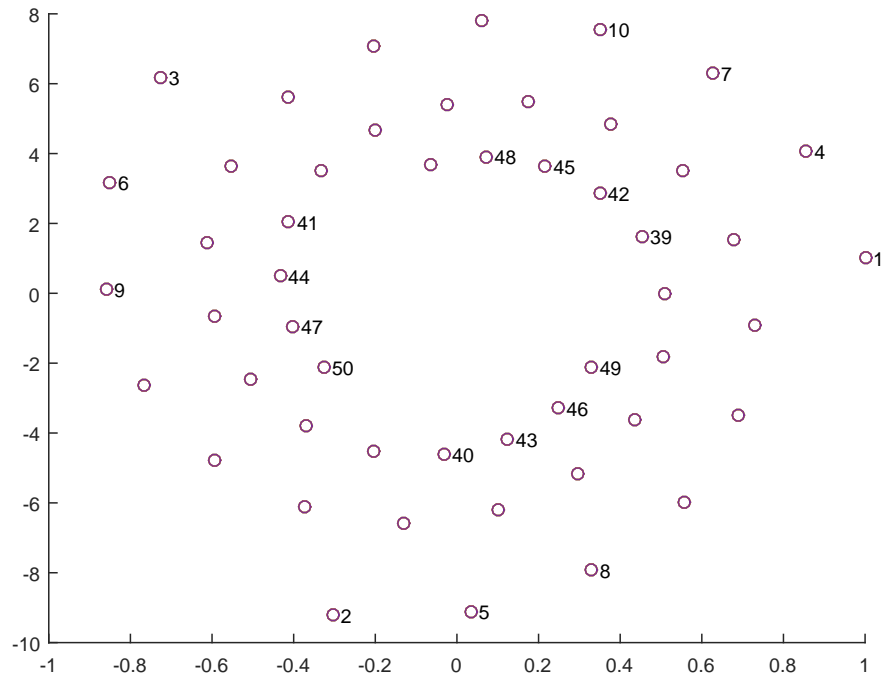


Figura 1.8: Espiralado hacia el origen al aplicar el método con la función test.

1.4. Estimación del error local y cambio de paso

Para la resolución numérica de ecuaciones diferenciales ordinarias es conveniente usar paso variable, h_n , para avanzar pasos pequeños donde la solución cambia rápidamente o pasos más grandes si la solución varía más lentamente. Para decidir cuál es la longitud del paso de integración más adecuada en cada momento, es preciso estimar de algún modo el error que se está cometiendo en la integración numérica. Dicha estimación se puede llevar a cabo mediante el uso de los llamados *pares encajados de métodos Runge-Kutta-Nyström*.

Un par encajado de métodos Runge-Kutta-Nyström consiste en dos métodos de órdenes p y \hat{p} (con $p > \hat{p}$) que comparten las mismas evaluaciones de función

$$\mathbf{k}_i = \mathbf{f}(\mathbf{y}_n + h_n \gamma_i \mathbf{v}_n + h_n^2 \sum_{j=1}^s \alpha_{ij} \mathbf{k}_j), \quad 1 \leq i \leq s.$$

En el modo de extrapolación local, en la integración numérica cada paso avanza con el método de orden más alto p

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h_n \mathbf{v}_n + h_n^2 \sum_{i=1}^s \beta_i \mathbf{k}_i, \quad (1.42)$$

$$\mathbf{v}_{n+1} = \mathbf{v}_n + h_n \sum_{i=1}^s b_i \mathbf{k}_i, \quad (1.43)$$

$$t_{n+1} = t_n + h_n,$$

y el método auxiliar de orden \hat{p}

$$\hat{\mathbf{y}}_{n+1} = \mathbf{y}_n + h_n \mathbf{v}_n + h_n^2 \sum_{i=1}^s \hat{\beta}_i \mathbf{k}_i, \quad (1.44)$$

$$\hat{\mathbf{v}}_{n+1} = \mathbf{v}_n + h_n \sum_{i=1}^s \hat{b}_i \mathbf{k}_i, \quad (1.45)$$

$$t_{n+1} = t_n + h_n,$$

se usa solamente para construir el estimador del error local (el error cometido por el método tras un único paso de integración)

$$E_{n+1} = \max(\|\mathbf{y}_{n+1} - \hat{\mathbf{y}}_{n+1}\|_\infty, \|\mathbf{v}_{n+1} - \hat{\mathbf{v}}_{n+1}\|_\infty). \quad (1.46)$$

Veamos que $\mathbf{y}_{n+1} - \hat{\mathbf{y}}_{n+1}$ y $\mathbf{v}_{n+1} - \hat{\mathbf{v}}_{n+1}$ son estimaciones de los errores locales cometidos por el método de orden \hat{p} . Para ello denotamos por $\tilde{\mathbf{y}}(t)$ la solución exacta de $\mathbf{y}''(t) = \mathbf{f}(\mathbf{y}(t))$ con condiciones iniciales $\tilde{\mathbf{y}}(t_n) = \mathbf{y}_n$ e $\tilde{\mathbf{y}}'(t_n) = \mathbf{v}_n$. Teniendo en cuenta que $\hat{p} < p$,

$$\begin{aligned} \mathbf{y}_{n+1} - \hat{\mathbf{y}}_{n+1} &= \mathbf{y}_{n+1} - \tilde{\mathbf{y}}(t_n + h_n) - (\hat{\mathbf{y}}_{n+1} - \tilde{\mathbf{y}}(t_n + h_n)) \\ &= \tilde{\mathbf{y}}(t_n + h_n) - \hat{\mathbf{y}}_{n+1} + O(h_n^{p+1}), \\ \mathbf{v}_{n+1} - \hat{\mathbf{v}}_{n+1} &= \mathbf{v}_{n+1} - \tilde{\mathbf{y}}'(t_n + h_n) - (\hat{\mathbf{v}}_{n+1} - \tilde{\mathbf{y}}'(t_n + h_n)) \\ &= \tilde{\mathbf{y}}'(t_n + h_n) - \hat{\mathbf{v}}_{n+1} + O(h_n^{p+1}), \end{aligned}$$

para h_n pequeño, de donde concluimos que, salvo términos $O(h_n^{p+1})$, E_{n+1} es el error local cometido por el método de orden bajo. Por otro lado, como

$$\begin{aligned} \tilde{\mathbf{y}}(t_n + h_n) - \hat{\mathbf{y}}_{n+1} &= C_1 h_n^{\hat{p}+1} + O(h_n^{\hat{p}+2}) = O(h_n^{\hat{p}+1}), \\ \tilde{\mathbf{y}}'(t_n + h_n) - \hat{\mathbf{v}}_{n+1} &= C_2 h_n^{\hat{p}+1} + O(h_n^{\hat{p}+2}) = O(h_n^{\hat{p}+1}), \end{aligned}$$

con C_1 y C_2 constantes, y para h_n pequeño $O(h_n^{p+1})$ es despreciable frente a $O(h_n^{\hat{p}+1})$, de estas igualdades obtenemos que

$$\mathbf{y}_{n+1} - \hat{\mathbf{y}}_{n+1} = O(h_n^{\hat{p}+1}), \quad \mathbf{v}_{n+1} - \hat{\mathbf{v}}_{n+1} = O(h_n^{\hat{p}+1}),$$

y, en consecuencia, $E_{n+1} \approx Ch_n^{\hat{p}+1}$ siendo C una constante.

Por otro lado, si la tolerancia TOL representa el máximo error local permitido, la longitud de paso óptima h_{opt} para avanzar un paso desde t_n generando un error local igual a la tolerancia debería satisfacer $TOL \approx Ch_{opt}^{\hat{p}+1}$. Compararemos el máximo error local permitido, TOL , con el error local estimado, E_{n+1} , para encontrar una forma de ajustar la longitud del paso de integración. De la igualdad

$$\frac{TOL}{E_{n+1}} = \left(\frac{h_{opt}}{h_n} \right)^{\hat{p}+1},$$

se deduce

$$h_{opt} = h_n \left(\frac{TOL}{E_{n+1}} \right)^{\frac{1}{\hat{p}+1}}.$$

Esto nos permite ajustar la longitud del paso de integración del siguiente modo

$$h_{opt} = 0.9h_n \left(\frac{TOL}{E_{n+1}} \right)^{\frac{1}{\hat{p}+1}}, \quad (1.47)$$

siendo 0.9 un *factor de seguridad* que se introduce para evitar en la medida de lo posible que se produzcan rechazos.

Si al intentar avanzar un paso de longitud h_n desde t_n a t_{n+1} se observa que $E_{n+1} \leq TOL$, entonces se acepta dicho paso y se calcula para avanzar el paso siguiente $h_{n+1} = h_{opt}$ dado por (1.47). Si, por el contrario, $E_{n+1} > TOL$, entonces el paso que se ha intentado para avanzar de t_n a t_{n+1} se rechaza, se calcula una nueva longitud de paso $h_n = h_{opt}$ con (1.47) y se repiten los cálculos (1.42)–(1.46) con el nuevo h_n .

Notamos que en la práctica no es necesario calcular la aproximación de orden \hat{p} , puesto que solamente son necesarias las diferencias

$$\mathbf{y}_{n+1} - \hat{\mathbf{y}}_{n+1} = h_n^2 \sum_{i=1}^s (\beta_i - \hat{\beta}_i) \mathbf{k}_i, \quad (1.48)$$

$$\mathbf{v}_{n+1} - \hat{\mathbf{v}}_{n+1} = h_n \sum_{i=1}^s (b_i - \hat{b}_i) \mathbf{k}_i, \quad (1.49)$$

que nos permiten hallar el error estimado y pueden ser calculadas como combinación lineal de las evaluaciones de función \mathbf{k}_i , $1 \leq i \leq s$, con los coeficientes apropiados $\beta_i - \hat{\beta}_i$ y $b_i - \hat{b}_i$, $1 \leq i \leq s$.

Capítulo 2

Integración numérica del problema de Kepler

En esta sección presentamos los resultados numéricos que se han obtenido al integrar con dos métodos Runge-Kutta-Nyström de órdenes 4 y 6, respectivamente, el conocido problema de Kepler. Ambos métodos se han implementado en Matlab tanto con paso fijo como con paso variable. En el Apéndice A se incluyen las correspondientes funciones.

2.1. Descripción del problema

El problema considerado para ilustrar el comportamiento de los métodos Runge-Kutta-Nyström estudiados en este trabajo es el problema de Kepler. Dicho problema describe el movimiento en un plano de un punto material que es atraído hacia el origen con una fuerza inversamente proporcional al cuadrado de su distancia al origen. En particular, podríamos pensar en un satélite que orbita alrededor de la Tierra.

Las ecuaciones del movimiento, cuando se formulan como un sistema diferencial de segundo orden, son

$$\mathbf{x}''(t) = \left(\frac{-x_1(t)}{\left(\sqrt{x_1(t)^2 + x_2(t)^2}\right)^3}, \frac{-x_2(t)}{\left(\sqrt{x_1(t)^2 + x_2(t)^2}\right)^3} \right)^T. \quad (2.1)$$

En nuestro estudio vamos a considerar condiciones iniciales de la forma

$$\mathbf{x}(0) = (1 - e, 0)^T, \quad \mathbf{x}'(0) = \left(0, \sqrt{\frac{1+e}{1-e}} \right)^T, \quad (2.2)$$

donde e es un parámetro ($0 < e < 1$).

Las soluciones de (2.1)–(2.2) corresponden a órbitas periódicas de periodo 2π que en el plano (x_1, x_2) describen elipses de excentricidad e y semieje mayor 1. A medida que va siendo mayor el valor de la excentricidad, se hace más necesario el uso de pasos variables en la integración numérica del problema porque a lo largo de la solución las velocidades varían enormemente. De hecho, el problema de Kepler es uno de los problemas test que se han utilizado desde los años 70 del siglo pasado de manera sistemática para analizar el buen comportamiento de los estimadores del error y de las implementaciones con paso variable de las distintas familias de integradores numéricos de ecuaciones diferenciales ordinarias.

2.2. Métodos utilizados

En los experimentos numéricos hemos utilizado los dos pares encajados de métodos Runge-Kutta-Nyström de órdenes 4 y 6, respectivamente, que se construyen en [3], atendiendo a criterios que garantizan una buena estimación de los errores locales y una eficiente implementación del cambio de paso. En ambos casos el método de orden mayor satisface la hipótesis simplificadora (1.28). Además, los dos métodos también satisfacen la propiedad conocida como FSAL (*first same as last*) esto es, la última etapa de cada paso coincide con la primera etapa del paso siguiente, es decir,

$$c_s = 1, \quad \beta_s = 0, \quad \alpha_{sj} = \beta_j, \quad j = 1, \dots, s - 1.$$

De este modo se ahorra una evaluación de función por cada paso aceptado.

El primer método considerado es un método de 4 etapas y de orden 4, con un método encajado de orden 3 para la estimación del error local, y cuyos coeficientes son los que se muestran en la Tabla 2.2. Al igual que en [3], nos referiremos a él como RKN4(3)4FM. La notación 4(3)4 se refiere a $p(\hat{p})s$, donde p denota el orden del método con el que vamos a hacer la integración, \hat{p} se refiere al orden del método con el que se estima el error local y s es el número de etapas del método. La letra F hace mención al hecho de que el método de orden alto satisface la propiedad FSAL y la letra M refleja que los coeficientes del método se han elegido para minimizar el término dominante del error local.

Se trata de un método que cumple la condición (1.29) y en el que el método de orden 4 cumple también la condición (1.28). Por tanto, para hallar los coeficientes de dicho método, sus autores tuvieron que resolver 10

0				
$\frac{1}{4}$	$\frac{1}{32}$			
$\frac{7}{10}$	$\frac{7}{1000}$	$\frac{119}{500}$		
1	$\frac{1}{14}$	$\frac{8}{27}$	$\frac{25}{189}$	
β_i	$\frac{1}{14}$	$\frac{8}{27}$	$\frac{25}{189}$	
b_i	$\frac{1}{14}$	$\frac{32}{81}$	$\frac{250}{567}$	$\frac{5}{54}$
$\hat{\beta}_i$	$\frac{-7}{150}$	$\frac{67}{150}$	$\frac{3}{20}$	$\frac{-1}{20}$
\hat{b}_i	$\frac{13}{21}$	$\frac{-20}{27}$	$\frac{275}{189}$	$\frac{-1}{3}$

Tabla 2.1: Coeficientes del método RKN4(3)4FM.

ecuaciones de orden (5 para \mathbf{v} , 3 para $\hat{\mathbf{v}}$ y 2 para $\hat{\mathbf{y}}$). La estimación del error local y el cambio de paso se han llevado a cabo como se indica en la Sección 1.4 con $s = 4$, $p = 4$ y $\hat{p} = 3$.

El segundo método que hemos considerado, es un método de 6 etapas y de orden 6, con un método encajado de orden 4 para la estimación del error local, cuyos coeficientes son los que se muestran en la Tabla 2.2. Nos referiremos a él, como en [3], por RKN6(4)6FM, puesto que $p = 6$, $\hat{p} = 4$ y $s = 6$, el método de orden 6 satisface la propiedad FSAL y sus coeficientes se obtuvieron minimizando los términos $O(h^7)$ del error local. Notamos que en este caso el orden del método con el que avanza la integración es dos unidades mayor que el orden del método que se utiliza para estimar el error local.

En este caso, se cumplen las condiciones (1.28) y (1.29) tanto para el método de orden 6 como para el de orden 4, por lo que los autores de [3] tuvieron que resolver 18 ecuaciones de orden (13 para \mathbf{v} y 5 para $\hat{\mathbf{v}}$) para hallar los coeficientes de este método. De nuevo, la estimación del error local y el cambio de paso se han realizado como se indica en la Sección 1.4 con $s = 6$, $p = 6$ y $\hat{p} = 4$.

0						
1	$\frac{1}{200}$					
3	$\frac{-1}{2200}$		$\frac{1}{22}$			
7	$\frac{637}{6600}$		$\frac{-7}{110}$		$\frac{7}{33}$	
17	$\frac{225437}{1968750}$		$\frac{-30073}{281250}$		$\frac{65569}{281250}$	
25	$\frac{-9367}{984375}$					
1	$\frac{151}{2142}$		$\frac{5}{116}$		$\frac{385}{1368}$	
	$\frac{55}{168}$		$\frac{-6250}{28101}$			
β_i	$\frac{1349}{157500}$		$\frac{7873}{50000}$		$\frac{-16}{125}$	
b_i	$\frac{1349}{157500}$		$\frac{7873}{45000}$		$\frac{-2}{5}$	
$\hat{\beta}_i$	$\frac{151}{2142}$		$\frac{5}{116}$		$\frac{385}{1368}$	
\hat{b}_i	$\frac{55}{168}$		$\frac{-6250}{28101}$		$\frac{1}{12}$	
	$\frac{151}{2142}$		$\frac{25}{522}$		$\frac{275}{684}$	
	$\frac{275}{252}$		$\frac{-78125}{112404}$		$\frac{1}{12}$	

Tabla 2.2: Coeficientes del método RKN6(4)6FM.

2.3. Resultados numéricos

Presentamos en esta sección los resultados numéricos que se han obtenido cuando el problema de Kepler (2.1)–(2.2) se ha integrado numéricamente con los dos métodos Runge-Kutta-Nyström descritos en la Sección 2.2, utilizando tanto paso fijo como paso variable. Se han considerado excentricidades $e = 0.3, 0.5$ y 0.7 , y el tiempo de integración ha sido $T = 30 \times 2\pi$ (es decir, 30 periodos).

En la implementación de paso fijo se han utilizado longitudes de paso $h = 2\pi/K$, $K = 2^k$, $5 \leq k \leq 11$, y para la implementación de paso variable las tolerancias consideradas han sido $TOL = 10^{-K}$, $K = 4, 5, 6, 7, 8, 9$. En todos los experimentos se han medido los errores al final de la integración donde, por la periodicidad de la solución, se sabe que $\mathbf{x}(T) = \mathbf{x}(0)$ y $\mathbf{x}'(T) = \mathbf{x}'(0)$. Se han medido errores tanto en la solución como en su derivada, utilizando para ello la norma euclídea de \mathbb{R}^4 .

Se han representado tres tipos de gráficas para visualizar los resultados obtenidos, todas ellas en escala doblemente logarítmica, es decir, en el eje de las x se representa el logaritmo del valor en lugar del propio valor y en el eje de las y ocurre lo mismo. Usamos este tipo de gráficas porque, como se explica más adelante, permiten visualizar más fácilmente algunas de las características de los métodos considerados, mediante la pendiente de las líneas que unen los datos representados. En todas las gráficas, para los datos obtenidos con la implementación de paso fijo utilizamos trazo continuo, mientras que para los generados con paso variable usamos el trazo discontinuo. Del mismo modo, representamos los resultados correspondientes al método de orden 4 en color azul y marcando los puntos representativos con círculos y los obtenidos con el método de orden 6 en color rojo y marcando los puntos representativos con $*$.

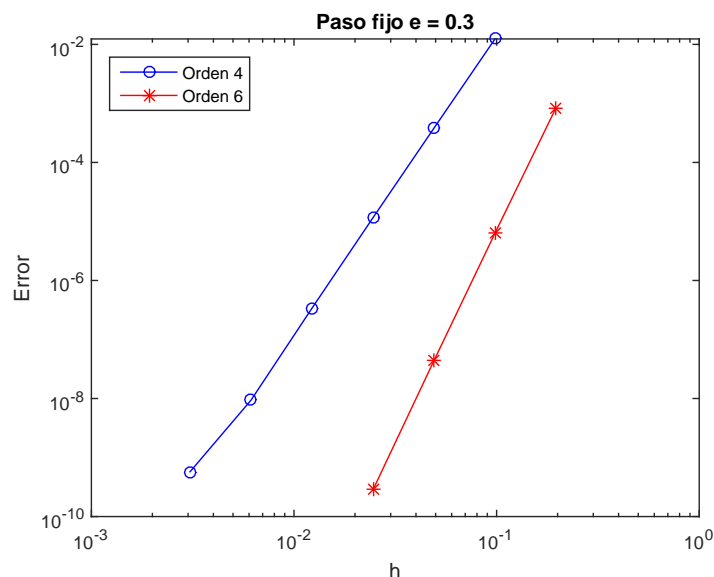


Figura 2.1: Gráficas referentes al paso fijo que representan el error frente a la longitud de paso con $exc = 0.3$.

Las gráficas incluidas en las Figuras 2.1 – 2.3 contienen los resultados obtenidos con paso fijo para excentricidades 0.3 (Figura 2.1), 0.5 (Figura 2.2) y 0.7 (Figura 2.3), respectivamente. En ellas, se ha representado el error frente a la longitud de paso de integración utilizado (en realidad, el logaritmo de ambos), para intentar visualizar gráficamente el orden de cada método. Si trabajamos con un método de orden p , el error global E (visto

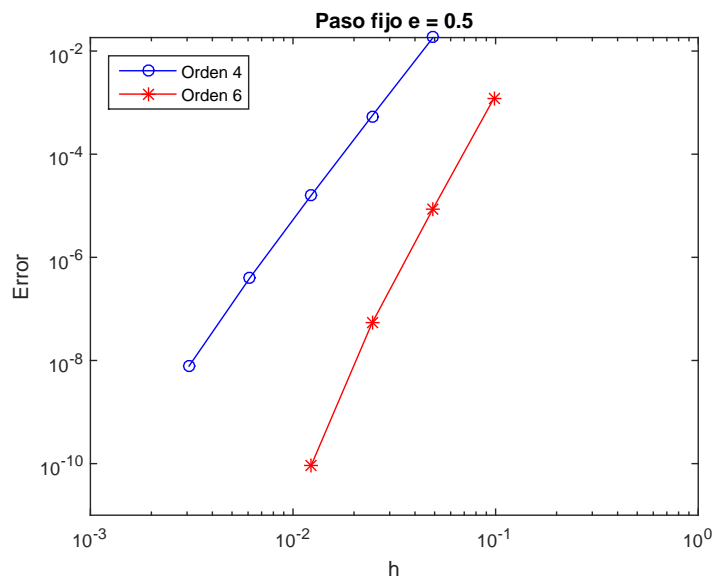


Figura 2.2: Gráficas referentes al paso fijo que representan el error frente a la longitud de paso con $\text{exc} = 0.5$.

como función de h) se comporta cuando h tiende a 0 como $E \approx Ch^p$. Tomando logaritmos es claro que $\log(E) \approx \log(C) + p \cdot \log(h)$, y al utilizar en las gráficas la escala doblemente logarítmica, representando $\log(E)$ frente a $\log(h)$, esperamos ver una línea recta de pendiente p .

En las tres gráficas observamos que la pendiente de las líneas rojas es aproximadamente 7 cuando esperaríamos solo pendiente 6, por tratarse de un método de orden $p = 6$. Análogamente, vemos que la pendiente de las líneas azules en lugar de ser 4, como correspondería a un método de orden $p = 4$, es próxima a 5. La explicación hay que buscarla en [1] donde se establece que para el problema de Kepler integrado con un método de un paso de orden p , el error tras N periodos se comporta como

$$NO(h^p) + \frac{N^2 - N}{2}O(h^{p+1}) + O(h^{p+2}). \quad (2.3)$$

Esto implica que fijado N , (en nuestro caso $N = 30$) el error visto como función de h se puede comportar como $O(h^{p+1})$ si el segundo sumando de (2.3) es mayor que el primero, o como $O(h^p)$, si el segundo sumando de (2.3) es menor que el primero. Esto justifica que las pendientes de las correspondientes líneas cuando se representa en escala doblemente logarítmica el error frente a h sea $p + 1$ en lugar de p , o una combinación de ambas.

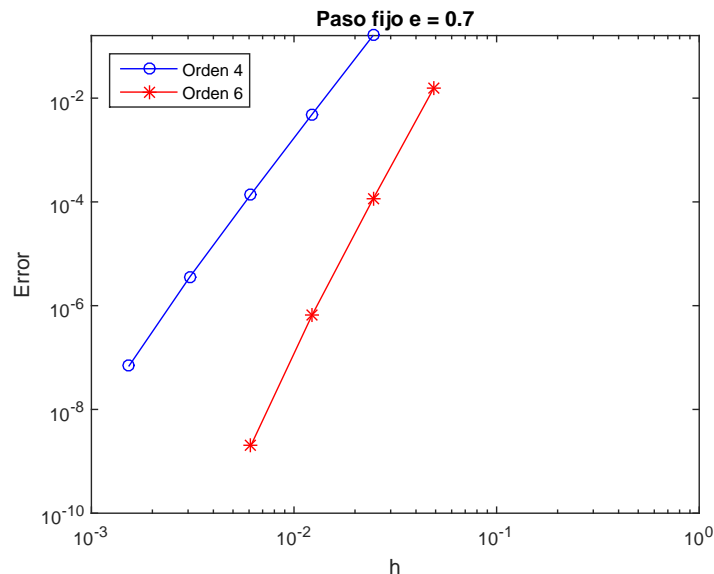


Figura 2.3: Gráficas referentes al paso fijo que representan el error frente a la longitud de paso con $exc = 0.7$.

En las Figuras 2.1, 2.2 y 2.3 hemos representado únicamente datos correspondientes a errores inferiores a 10^{-1} . En el caso del método de orden 4 esto implica que el paso más grande representado para excentricidad 0.3 es $h = 2\pi/32$, mientras que para excentricidades 0.5 y 0.7 ha sido necesario empezar con $h = 2\pi/128$ y $h = 2\pi/256$, respectivamente, para conseguir un error del mismo tamaño.

Con el método de orden 6, para excentricidad 0.3 el mayor paso utilizado es $h = 2\pi/32$, mientras que para excentricidad 0.5 necesitamos $h = 2\pi/64$ y para la excentricidad 0.7 $h = 2\pi/128$, para conseguir un error por debajo del 10%. Se observa que a medida que aumenta la excentricidad de la órbita aumenta también la dificultad de la integración de este problema, un hecho que ya anunciábamos en la Sección 2.1.

Comparando ahora para cada excentricidad los resultados obtenidos con el método de orden 4 y el de orden 6 vemos que para el mismo h , los errores con el método de orden 6 son menores que con el método de orden 4, como corresponde al comportamiento cuando h tiende a 0 de $O(h^6)$ frente a $O(h^4)$.

Las gráficas incluidas en las Figuras 2.4 – 2.6 contienen los resultados obtenidos con ambos métodos, implementados con paso variable, para las mismas excentricidades 0.3, 0.5 y 0.7. Como se ha indicado anteriormente,

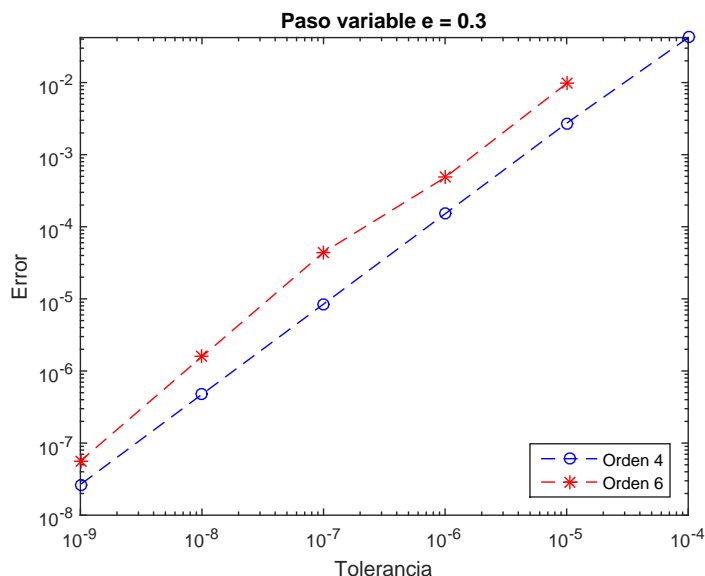


Figura 2.4: Gráficas referentes al paso variable que representan el error frente a la tolerancia con $\text{exc} = 0.3$.

en cada gráfica la línea azul corresponde al método de orden 4 y la roja al método de orden 6. Se ha representado ahora el error frente a la tolerancia utilizada (en realidad, el logaritmo de ambos) para intentar observar si el error global obtenido es proporcional a dicha tolerancia. Si así fuese se esperarían líneas de pendiente 1 en escala doblemente logarítmica puesto que si $E \approx C \cdot TOL$, tomando logaritmos obtenemos $\log(E) \approx \log(C) + \log(TOL)$. No obstante, como en la implementación de paso variable considerada h_{opt} se elige estimando el error local, de modo que $TOL = h_{opt}^{\hat{p}+1}$, y el error global se comporta como $E \approx \max_n h_n^p$, lo que se espera es $E \approx C \cdot TOL^{p/(\hat{p}+1)}$ y, en consecuencia, $\log(E) \approx \log(C) + (p/(\hat{p}+1)) \cdot \log(TOL)$. Esto corresponde a una recta de pendiente $p/(\hat{p}+1)$ que en el caso del método de orden 4 tomaría el valor 1, pero que para el método de orden 6 tomaría el valor $\frac{6}{5}$. Sin embargo, las pendientes observadas en las Figuras 2.4–2.6 están más cerca de $\frac{5}{4}$ (azul) y $\frac{7}{5}$ (rojo), lo que corresponde a $\frac{p+1}{\hat{p}+1}$, que está de acuerdo con el hecho ya observado en las Figuras 2.1 – 2.3 de que el error global para este problema se está comportando como $O(h^{p+1})$ en vez del esperado $O(h^p)$.

De nuevo se han representado solamente datos correspondientes a errores

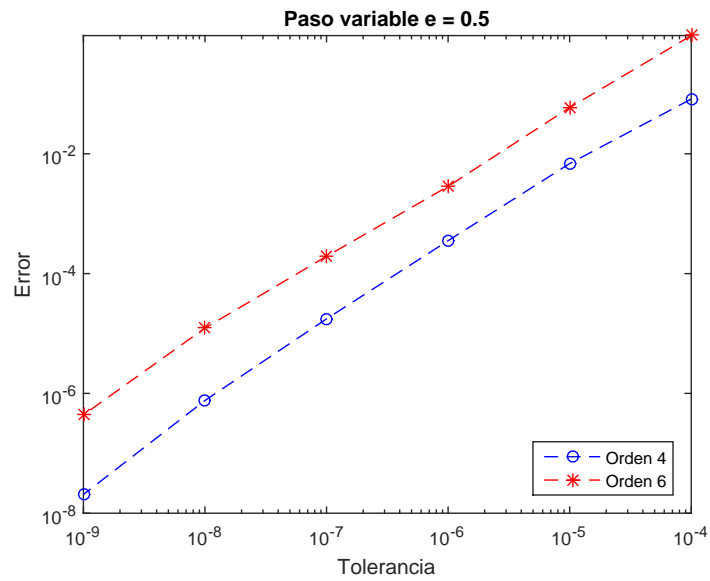


Figura 2.5: Gráficas referentes al paso variable que representan el error frente a la tolerancia con $\text{exc} = 0.5$.

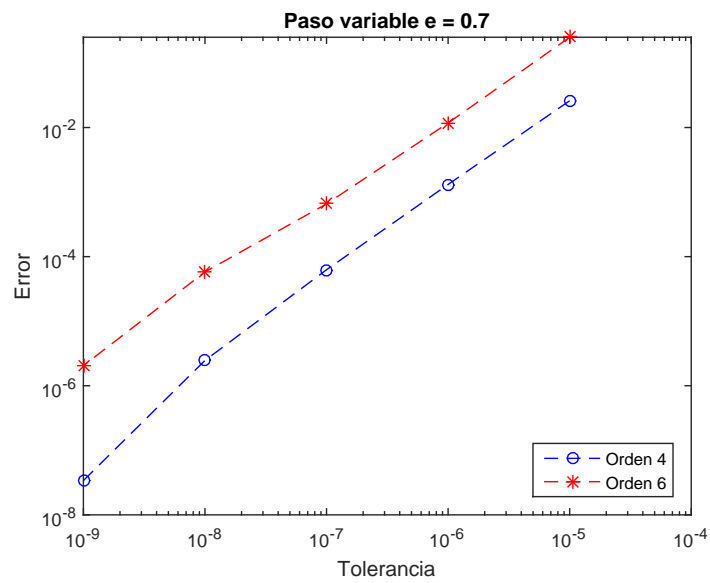


Figura 2.6: Gráficas referentes al paso variable que representan el error frente a la tolerancia con $\text{exc} = 0.7$.

inferiores a 10^{-1} . Para ambos métodos observamos que para excentricidades 0.3 y 0.5 la mayor tolerancia utilizada es $TOL = 10^{-4}$, mientras que para excentricidad 0.7 ha sido necesario empezar con $TOL = 10^{-5}$, para conseguir un error inferior al 10%.

Igual que hicimos con la implementación de paso fijo, si comparamos para cada excentricidad los resultados obtenidos con el método de orden 4 y con el de orden 6, en este caso, vemos que para una misma tolerancia los errores con el método de orden 4 son menores ligeramente que con el método de orden 6.

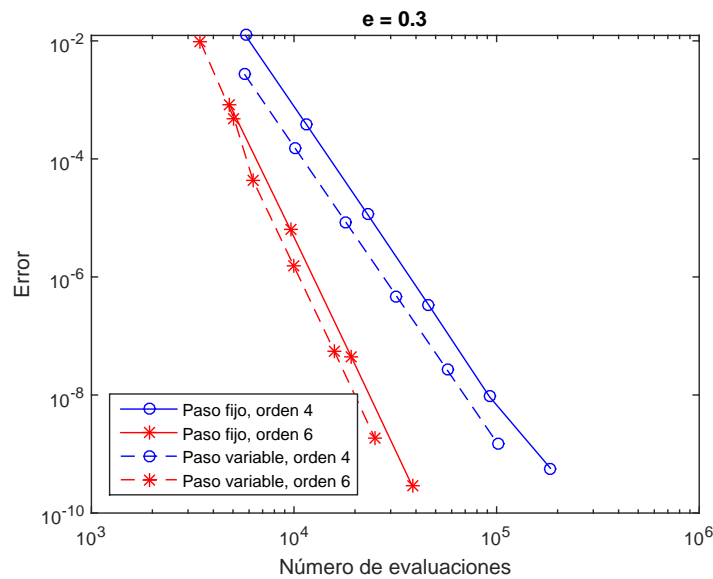


Figura 2.7: Gráficas de eficiencia que representan error frente al número de evaluaciones de función con $exc = 0.3$.

Por último, hemos representado en las Figuras 2.7, 2.8, 2.9 las gráficas de eficiencia, error frente a número de evaluaciones del lado derecho de (2.1) para ver cual de las 4 implementaciones consideradas es la más eficiente, es decir, para un mismo error, necesita un menor número de evaluaciones de función. En este caso, representamos conjuntamente los datos obtenidos con paso fijo y con paso variable. En las Figuras 2.7 – 2.9 observamos que las líneas que corresponden al método de orden 4 tienen pendiente -5 y que las que hacen referencia al método de orden 6 tienen pendiente -7 , que está de acuerdo con (2.3) y con el hecho de que el número de evaluaciones de función es proporcional al número de pasos dados y, por tanto, inversamente

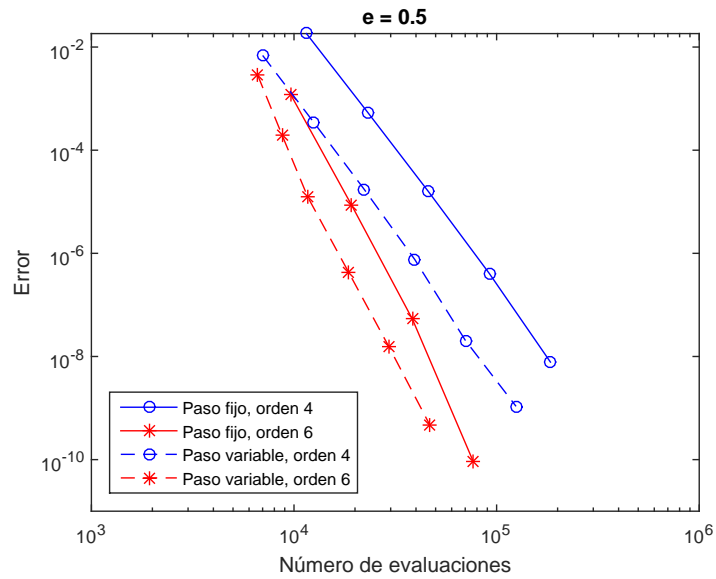


Figura 2.8: Gráficas de eficiencia que representan error frente al número de evaluaciones de función con $\text{exc} = 0.5$.

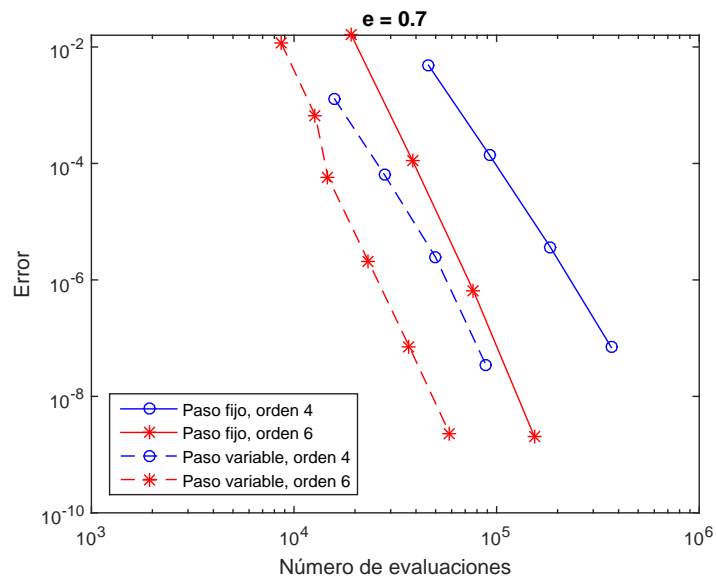


Figura 2.9: Gráficas de eficiencia que representan error frente al número de evaluaciones de función con $\text{exc} = 0.7$.

proporcional a la longitud del paso de integración h . Notamos que para la excentricidad más baja la eficiencia de paso variable y de paso fijo es muy similar, pero a medida que la excentricidad va aumentando se puede ver claramente como las implementaciones de paso variable son más eficientes que las de paso fijo.

Para excentricidad 0.3, tanto para el método de orden 4 como para el método de orden 6, hay poca diferencia entre la implementación de paso fijo y la de paso variable aunque es un poco más eficiente esta última pues para obtener un error dado requiere un número de evaluaciones ligeramente menor. Sin embargo, en la Figura 2.9 observamos que con excentricidad 0.7 la implementación de paso variable es claramente más eficiente que la de paso fijo. Por ejemplo, para obtener un error de tamaño 10^{-7} el método de orden 4 implementado con paso fijo requiere 368.644 evaluaciones de función mientras que implementado con paso variable la misma precisión se consigue con 88.792 evaluaciones. El costo computacional medido en número de evaluaciones de función se divide por más de 4 al utilizar pasos variables para este valor alto de la excentricidad. Para excentricidad 0.5, aunque la ventaja de pasar de paso fijo a variable no es tan grande, también observamos que, por ejemplo, para obtener un error de tamaño 10^{-7} con el método de orden 4 la implementación de paso fijo requiere aproximadamente el doble de evaluaciones de función que la de paso variable.

Un comentario análogo se puede hacer cuando se comparan las dos implementaciones del método de orden 6. Para excentricidad 0.3 no se aprecia gran diferencia a la hora de implementar paso fijo o paso variable, pero cuando se trabaja con excentricidad 0.7, para obtener un error de tamaño 10^{-5} , se necesitan 23.346 evaluaciones de función en paso variable, mientras que para el paso fijo se multiplica por más de 3 el número de evaluaciones, siendo 76.801 las utilizadas.

Si tratamos ahora de identificar cuál de las 4 implementaciones consideradas es la más eficiente, veremos que en todos los casos corresponde a la implementación de paso variable del método de orden 6.

Bibliografía

- [1] Calvo, M. P., *Métodos Runge-Kutta-Nyström simplécticos*. Tesis Univ. de Valladolid, Facultad de Ciencias, (1992).
- [2] Calvo, M. P., *Nyström methods*. Encyclopedia of Applied and Computational Mathematics, Springer, 1080-1087 (2015).
- [3] Dormand, J. R., El-Mikkawy, M. E. A., Prince, P. J., *Families of Runge-Kutta-Nystrom Formulae*. IMA Journal of Numerical Analysis 7, 235-250 (1987).
- [4] Galindo, F., Sanz, J., Tristán, L. A., *Cálculo infinitesimal en una variable*. Thomson, (2003).
- [5] Hairer, E., Norsett, S. P., Wanner, G., *Solving ordinary differential equations. I, Nonstiff problems*. Springer, Berlin (1993).

Apéndice A

Programas de Matlab

En este Apéndice mostramos el código de los programas de Matlab que hemos realizado para la implementación de los métodos de paso fijo y variable así como del estudio de la estabilidad absoluta y la periodicidad.

A.1. Implementación de paso fijo

En la función `metodo` calculamos la aproximación a la solución con paso fijo. Es una función general que sirve para integrar cualquier sistema diferencial de segundo orden de la forma (1.1), no sólo para el problema de Kepler.

```
function [tt,yy,vv,neval] = metodo(t0,tF,y0,v0,h,fun,ind);
% t0 y tF son el tiempo inicial y el final
% y0 y v0 son las condiciones iniciales
% h es la longitud de paso de integracion
% fun es el nombre de la funcion que evalua el lado derecho
%           del sistema de segundo orden
% ind es para elegir cual de los dos metodos queremos usar
t = t0;
y = y0;
v = v0;
tt = [t];
yy = [y'];
vv = [v'];
% Cargamos los coeficientes del metodo elegido
[A, c, b, bb, e, ee, s, p, ph]=coeficientes(ind);
d = length(y0);
k = zeros(d,s);
```

```

k(:,1) = fun(t,y); % primera etapa
neval = 1;
h2 = h*h;
while t<tF
    if t+h>tF
        h=tF-t;
        h2 = h*h;
    end
    for i = 2:s; % calculamos el resto de las etapas del metodo
        k(:,i) = fun(t + c(i)*h, y + c(i)*h*v + h2*k*A(i,:));
        neval=neval+1;
    end
    y = y + h*v + h2*k*bb;
    v = v + h*k*b;
    t=t+h;
    k(:,1) = k(:,s); % Propiedad FSAL
    tt=[tt; t];
    yy=[yy; y'];
    vv=[vv; v'];
end
end

```

En la función `metodokepler` calculamos la aproximación a la solución de (2.1)–(2.2) con paso fijo para diferentes longitudes de paso llamando a la función `metodo`. También dibujamos el error frente a la longitud de paso y el error frente al número de evaluaciones.

```

function [error,hh,neval] = metodokepler(e,N,K,nrep,ind);
% e es la excentricidad de la orbita
% N es el numero de periodos durante los que vamos a integrar
% K es tal que la longitud de paso inicial es 2*pi/K
% nrep es el numero de veces que se divide por dos el paso inicial
% ind es para indicar el metodo que queremos usar
t0 = 0; tF = 2*N*pi; % tiempo inicial y final
y0 = [1-e; 0]; % condicion inicial
v0 = [0; sqrt((1+e)/(1-e))]; % condicion inicial para la derivada
h = 2*pi/K; % longitud de paso utilizado en la primera integracion
hh = zeros(1, nrep);
neval = zeros(1, nrep);
for i=1:nrep;
    [t, y, v, neval(i)]=metodo(t0, tF, y0, v0, h, @kepler, ind);
    % llamada a la funcion del metodo de paso fijo.
    n = length(t);
    error(i) = norm([y(n,:) , v(n,:)']-[y0,v0]);
    hh(i) = h;
    h = h/2;
end

```

```

end
figure(1)
loglog(hh,error,'o-')
xlabel('h')
ylabel('Error')
figure(2)
loglog(neval,error,'o-')
xlabel('Numero de evaluaciones')
ylabel('Error')
end

```

A.2. Implementación con paso variable

En la función `metodovar` calculamos la aproximación a la solución con paso variable. Al igual que la función `metodo`, esta función es una función general que sirve para integrar con paso variable cualquier sistema diferencial de segundo orden de la forma (1.1).

```

function [tt,yy,vv,nev] = metodovar(t0,tF,y0,v0,tol,fun,ind);
% t0 y tF son el tiempo inicial y el final
% y0 y v0 son las condiciones iniciales
% tol es la tolerancia utilizada
% fun es el nombre de la funcion que evalua el lado derecho
%           del sistema de segundo orden
% ind es para elegir cual de los dos metodos queremos usar
t = t0;
y = y0;
v = v0;
tt = [t];
yy = [y'];
vv = [v'];
[A, c, b, bb, e, ee, s, p, ph] = coeficientes(ind);
d = length(y0);
k = zeros(d,s);
k(:,1) = fun(t,y); % primera etapa
neval = 1; % numero de evaluaciones
h=tol^(1/(ph+1)); % longitud de paso inicial
h2 = h*h;
nstep = 0; % numero de pasos dados
nreject = 0; % numero de pasos rechazados
nev = zeros(1,3);
while t<tF
    nstep = nstep + 1;
    if t+h>tF

```

```

        h = tF-t;
        h2 = h*h;
    end
    for i = 2:s; % resto de etapas
        k(:,i) = fun(t + c(i)*h, y + c(i)*h*v + h2*k*A(i,:));
        neval = neval+1;
    end
    errory = h2*k*ee;
    errorv = h*k*e;
    error = norm([errory;errorv]);
    if error < tol
        y = y + h*v + h2*k*bb;
        v = v + h*k*b;
        t = t+h;
        k(:,1) = k(:,s); % Propiedad FSAL
        tt = [tt; t];
        yy = [yy; y'];
        vv = [vv; v'];
        h = 0.9*h*(tol/error)^(1/(ph+1));
        h2 = h*h;
    else
        nreject = nreject + 1;
        h = 0.9*h*(tol/error)^(1/(ph+1));
        h2 = h*h;
    end
end
nev = [neval nreject nstep];
end

```

En la función `metodokeplervar` calculamos la aproximación a la solución de (2.1)–(2.2) con paso variable para diferentes tolerancias llamando a la función `metodovar`. Dibujamos también el error frente a la tolerancia y el error frente al número de evaluaciones.

```

function [error,toler,neval]=metodokeplervar(e,N,toll,nexp,ind);
% e es la excentricidad de la orbita
% N es el numero de periodos durante los que vamos a integrar
% toll es la tolerancia utilizada en la primera integracion
% nexp es el numero de veces que se divide la tolerancia por diez
% ind es para indicar el tipo de metodo que queremos usar
t0 = 0; tF = 2*N*pi; % tiempo inicial y final
y0 = [1-e; 0]; % condicion inicial
v0 = [0; sqrt((1+e)/(1-e))]; % condicion inicial para la derivada
tol = toll; % tolerancia utilizada en la primera integracion
nev = zeros(nexp,3);
toler = zeros(1,nexp);

```

```

for i=1:nexp;
    [t,y,v,nev(i,:)] = metodovar(t0,tF,y0,v0,tol,@kepler,ind);
    H=t(2:end)-t(1:end-1); % longitudes de paso utilizadas
    n = length(t);
    error(i) = norm([y(n,:) ',v(n,:) ']-[y0,v0]);
    toler(i) = tol;
    tol = tol/10;
end
figure(1)
loglog(toler,error,'o-')
xlabel('Tolerancia')
ylabel('Error')
figure(2)
loglog(nev(:,1),error,'o-')
xlabel('Numero de evaluaciones')
ylabel('Error')
neval = nev(:,1)';
end

```

A.3. Estabilidad del método RKN4(3)4FM

En la función `estabilidad1` estudiamos la región de estabilidad absoluta del método Runge-Kutta-Nyström de orden 4 y dibujamos las gráficas de las Figuras 1.5, 1.6 y 1.7.

```

function estabilidad1(ind)
[A, c, b, bb, e, ee, s, p, ph]=coeficientes (ind);
z = [-16:0.01:0];
r = length(z);
x = zeros(1,2);
e = ones(s,1);
e2 = ones(r,1);
bb = bb';
b = b';
for i=1:r
v = eye(s) - z(i)*A;
x = [v\e v\c];
% matriz de estabilidad
M = [1+z(i)*bb*x(:,1) 1+z(i)*bb*x(:,2);
      z(i)*b*x(:,1) 1+z(i)*b*x(:,2)];
S(i) = trace(M);
P(i) = det(M);
Q(i) = S(i)^2 - 4* P(i); % discriminante

```

```

rr(:,i) = abs(eig(M));
rho(i) = max(abs(eig(M))); % radio espectral de M
end
figure(1)
clf
plot(z,e2,'k',z,rr(1,:), 'r',z,rr(2,:), 'c',z,rho,'g')
axis([-16 0 0 1.6])
legend('y = 1', '\mu_1', '\mu_2', '\rho(M(z))')
legend('Location', 'southeast')
legend('boxoff')
limite = e2'+P;
figure(2)
clf
plot(z,abs(P),z,abs(S),z,e2,z,limite,z,Q)
legend('|P(z)|', '|S(z)|', 'y = 1', 'P(z) + 1', 'S(z)^2 - 4P(z)')
axis([-16 0 -4 6])
legend('Location', 'northeast')
legend('boxoff')
hold on
% Dibujo de caja de zoom
plot([-16,-12],[-0.6,-0.6], 'k:', [-16,-12],[1.2,1.2], 'k:')
plot([-12,-12],[-0.6,1.2], 'k:')
end

```

En la función `estabilidad2` dibujamos la gráfica de la Figura 1.8.

```

function estabilidad2(ind)
[A, c, b, bb, e, ee, s, p, ph]=coeficientes (ind);
x = zeros(1,2);
bb = bb';
b = b';
rr = zeros(2,1);
e = ones(s,1);
maxx = 50;
omega = 10;
h = 0.2;
z=-(h^2*omega^2);
y0 = 1;
hv0 = h;
y(1) = y0;
hv(1) = hv0;
AUX=[y(1); hv(1)];
v = eye(s) - z*A;
x = [v\e v\c];
M = [1+z*bb*x(:,1) 1+z*bb*x(:,2); z*b*x(:,1) 1+z*b*x(:,2)];
for j = 1:maxx
    AUX = M*AUX;

```



```

        y(j+1) = AUX(1, :);
        hv(j+1) = AUX(2, :);
    end
    vv = hv/h;
    hold on
    plot(y, vv, 'o')
    for j = 1:10
        text(y(j)+0.02, vv(j)+0.02, num2str(j))
    end
    for j = maxx-11:maxx
        text(y(j)+0.02, vv(j)+0.02, num2str(j))
    end
end
end

```

A.4. Otras funciones auxiliares

En la función `coeficientes` guardamos los coeficientes de los diferentes métodos Runge-Kutta-Nyström utilizados.

```

function [A, c, b, bb, e, ee, s, p, ph] = coeficientes(ind);
switch (ind)
case 1 % RKN4(3) 4FM
    A = [0 0 0 0; 1/32 0 0 0;
          7/1000 119/500 0 0; 1/14 8/27 25/189 0];
    c = [0; 1/4; 7/10; 1];
    b = [1/14; 32/81; 250/567; 5/54];
    bb = [1/14; 8/27; 25/189; 0];
    bh = [13/21; -20/27; 275/189; -1/3];
    bbh = [-7/150; 67/150; 3/20; -1/20];
    s = 4;
    p = 4;
    ph = 3;
case 2 % RKN6(4) 6FM
    A = [0 0 0 0 0 0;
          1/200 0 0 0 0;
          -1/2200 1/22 0 0 0;
          637/6600 -7/110 7/33 0 0;
          225437/1968750 -30073/281250 65569/281250 -9367/984375 0 0;
          151/2142 5/116 385/1368 55/168 -6250/28101 0];
    c = [0; 1/10; 3/10; 7/10; 17/25; 1];
    b = [151/2142; 25/522; 275/684; 275/252; -78125/112404; 1/12];
    bb = [151/2142; 5/116; 385/1368; 55/168; -6250/28101; 0];
    bh = [1349/157500; 7873/45000; 27457/90000;
          521683/630000; -2/5; 1/12];

```

```
bbh = [1349/157500; 7873/50000; 192199/900000;  
       521683/2100000; -16/125; 0];  
s = 6;  
p = 6;  
ph = 4;  
end  
ee = bb - bbh;  
e = b - bh;  
end
```

En la función `kepler` evaluamos el lado derecho de (2.1) que corresponde a la formulación del problema de Kepler como sistema de segundo orden.

```
function f = kepler(t,y);  
r=(sqrt(y(1)*y(1)+y(2)*y(2)))^3;  
f=[-y(1)/r; -y(2)/r]; % Problema de Kepler  
end
```

Apéndice B

Caracterización de la región de estabilidad absoluta

En este Apéndice demostraremos la equivalencia entre (1.39) y (1.40). Como $M(z)$ es una matriz 2×2 , los autovalores de $M(z)$ son las raíces μ_1, μ_2 de la ecuación característica

$$\mu^2 - b\mu + c = 0,$$

donde $b = P(z)$ y $c = S(z)$, siendo $P(z)$ el determinante de $M(z)$ y $S(z)$ su traza. Demostraremos que para $b, c \in \mathbb{R}$

$$|\mu_i| < 1 \quad i = 1, 2 \iff |c| < 1 \quad y \quad |b| < 1 + c,$$

teniendo en cuenta que

$$\mu_1 \cdot \mu_2 = c, \quad \mu_1 + \mu_2 = b. \tag{B.1}$$

Demostración. Vamos a demostrar cada una de las dos implicaciones.

\Rightarrow] Supongamos en primer lugar que $|\mu_i| < 1 \quad i = 1, 2$.

Como se cumple que $\mu_1 \cdot \mu_2 = c$ entonces $|c| < 1$.

Para la segunda condición separaremos la demostración en tres casos dependiendo de la naturaleza de las raíces μ_1 y μ_2 .

- Si μ_1 y μ_2 son un par de raíces complejas conjugadas, es decir, $\mu_1 = \alpha + i\beta$ y $\mu_2 = \alpha - i\beta$, con $\beta \neq 0$, entonces usando (B.1), obtenemos que $b = 2\alpha$ y $c = \alpha^2 + \beta^2 < 1$.

Ahora bien,

$$1 + c = 1 + \alpha^2 + \beta^2 = (1 - \alpha)^2 + 2\alpha + \beta^2 > 2\alpha = b$$

Análogamente,

$$-(1+c) = -1 - \alpha^2 - \beta^2 = -(1+\alpha)^2 + 2\alpha - \beta^2 < 2\alpha = b$$

Teniendo en cuenta lo anterior podemos concluir que $|b| < 1+c$.

- Si μ_1 y μ_2 son una raíz real doble, es decir, $\mu_1 = \mu_2 = \alpha$, entonces, usando (B.1), obtenemos que $b = 2\alpha$ y $c = \alpha^2 < 1$.

Ahora bien,

$$1+c = 1 + \alpha^2 = (1-\alpha)^2 + 2\alpha > 2\alpha = b$$

Del mismo modo,

$$-(1+c) = -1 - \alpha^2 = -(1+\alpha)^2 + 2\alpha < 2\alpha = b$$

Luego se concluye que $|b| < 1+c$.

- Si μ_1 y μ_2 son dos raíces reales diferentes, es decir, $\mu_1 = \alpha$ y $\mu_2 = \beta$, con $\alpha \neq \beta$, por (B.1), obtenemos $c = \alpha\beta$ y $b = \alpha + \beta$, como $|\alpha|, |\beta| < 1$ entonces tenemos que $(\alpha-1), (\beta-1) < 0$ y por tanto,

$$(\alpha-1)(\beta-1) = \alpha\beta - \alpha - \beta + 1 > 0$$

$$1 + \alpha\beta > \alpha + \beta$$

$$1 + c > b$$

Análogamente,

$$-(\alpha+1)(\beta+1) = -\alpha\beta - \alpha - \beta - 1 < 0$$

$$-1 - \alpha\beta < \alpha + \beta$$

$$-1 - c < b$$

Concluimos entonces que $|b| < 1+c$.

⇐] Partimos ahora de que $|c| < 1$ y $|b| < 1+c$, $b, c \in \mathbb{R}$.

Del mismo modo, separaremos la demostración en tres casos dependiendo de la naturaleza de las raíces μ_1 y μ_2 .

- Si μ_1 y μ_2 son un par de raíces complejas conjugadas, es decir, $|\mu_1| = |\mu_2|$ y teniendo en cuenta (B.1), obtenemos que $|\mu_1|^2 = c < 1$.

- Si μ_1 y μ_2 son una raíz real doble, es decir, $\mu_1 = \mu_2$, entonces por causa de (B.1), $\mu_1^2 = c$, $|\mu_1| = |\mu_2| = \sqrt{c} < 1$.
- Si μ_1 y μ_2 son dos raíces reales diferentes, es decir, $\mu_1 = \frac{1}{2}(b + \sqrt{b^2 - 4c})$ y $\mu_2 = \frac{1}{2}(b - \sqrt{b^2 - 4c})$. Luego,

$$|\mu_i| < \frac{1}{2}|b| + |\sqrt{b^2 - 4c}|.$$

Como $b^2 - 4c < (1 + c)^2 - 4c < 1 + c^2 - 2c = (1 - c)^2$, tenemos que

$$\frac{1}{2}|b| + |\sqrt{b^2 - 4c}| < \frac{1}{2}((1 + c) + (1 - c)) = 1$$

Por tanto, $|\mu_i| < 1$, $i = 1, 2$.

□

